In presenting the dissertation as a partial fulfillment of the requirements for an advanced degree from the Georgia Institute of Technology, I agree that the Library of the Institute shall make it available for inspection and circulation in accordance with its regulations governing materials of this type. I agree that permission to copy from, or to publish from, this dissertation may be granted by the professor under whose direction it was written, or, in his absence, by the Dean of the Graduate Division when such copying or publication is solely for scholarly purposes and does not involve potential financial gain. It is understood that any copying from, or publication of, this dissertation which involves potential financial gain will not be allowed without written permission.

7/25/68

FORECASTING PROCEDURES IN THE
TWO-ECHELON INVENTORY PROBLEM

A THESIS

Presented to

The Faculty of the Division of Graduate

Studies and Research

by

Robert Herron Wood

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in the School of Industrial Engineering

Georgia Institute of Technology

June, 1972

FORECASTING PROCEDURES

IN THE TWO-ECHELON INVENTORY PROBLEM

Approved:

Lynwood A. Johnson, Chairman

Douglas C. Montgomery

Richard H. Deane

Date approved by Chairman: MAY 30, 1972

ACKNOWLEDGMENTS

I would like to express my appreciation to those who have contributed so freely their time and knowledge to this research. I am deeply indebted to Professor Lynwood A. Johnson for the invaluable guidance he has provided. Through all stages in the development of this thesis, his expert knowledge has carried me by many obstacles. I would also like to gratefully acknowledge my readers, Professors Douglas C. Montgomery and Richard H. Deane. Their valuable suggestions have assisted me greatly. A special thanks is due Professor Deane for his help in preparing the final draft.

To my wife, Kathleen, I extend my deepest appreciation. Without her understanding, encouragement and patience, this work could not have been completed.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

SUMMARY

The performance of three alternate procedures for forecasting second-echelon demand in a two-echelon inventory system are examined. Simulation tests are conducted to yield results for a comparison of these performances when different input demand processes and forecasting models are used. Performance is measured by the values of unit days of inventory and unit days of backorders obtained in the tests.

The three second-echelon forecasting procedures studied in the research are:

1. Forecasts at the second echelon are based upon orders placed by the first-echelon branches.

2. Forecasts at the second echelon are based upon the customer demand experienced by the first-echelon branches.

3. Forecasts at the second echelon are based upon the forecasts made by the first-echelon branches.

Results from the tests show that the comparative performance of the three procedures is highly sensitive to the input demand processes and forecasting models used. That procedure which produces the smallest values of unit days of inventory generally produces the largest values of unit days of backorders. Performances with Procedure 1 in these tests are poor in comparison with those of the other two procedures. While values of backorders with Procedure 1 are uniformly lowest, this is offset by the large values of inventory stocks that result. Use of

Procedure 2 with a seasonal forecasting model yields consistently good results. No other preference can be stated between Procedures 2 and 3 for a particular demand process or for a particular forecasting model. Little distinction can be made between the performances of these latter two procedures; on no test is the difference in their values of unit days of inventory greater than 8 per cent; their values of unit days of backorders are close in all tests, except those with a seasonal forecasting model.

CHAPTER I

INTRODUCTION

The problem of sales (demand) forecasting for the control of single-stage inventory systems has been extensively examined by modern researchers; however, only recently has consideration been given to the problems encountered when forecasting for the more complex multi-echelon type of inventory system. From results reported to date, it appears evident that the following two factors have an important bearing on these problems. The first of these is the length and complexity of delays encountered as multi-echelon systems respond to inputs. The second concerns the growing distortion as the input signals travel through succeeding levels away from the point of entry. No new forecasting methods, however, have been suggested; rather, concern has centered on how existing methods can be used in the multi-echelon system in light of the above factors.

Forecasting at the lowest echelon presents no difficulties since the input signals to this level are the external system inputs, free of the distortion of management policy. However, at succeeding echelons there are alternative ways of defining the input information for forecasting. The following are three possible procedures:

1. Forecasts at echelons higher than the first are based on order quantities requested from facilities supported in the next lower echelon.

2. Information of customer demands (inputs at the lowest echelon) is transmitted to all levels within the system, and all forecasts are based upon this data.

3. Only the first-echelon facilities make forecasts. All other forecasts are achieved by combining these first-echelon forecasts.

There is no reason to believe that these three procedures (information modes) will result in the same system performance, since different delays and degrees of distortion are involved in each.

Purpose of the Research

The objective of this research is to gain a better understanding of how alternate modes of input information for forecasting at higher echelons impact upon system performance. Specifically, the three ways of defining input information which were presented in the preceding paragraph have been evaluated with regard to their impact on total inventory carried and total stock-outs generated. This investigation should represent a foundation study for a continuing research into this key aspect of multi-echelon forecasting.

Research Procedure

In pursuit of the above objective, a simple two-echelon inventory system has been modelled and then analyzed through computer simulations. In the analysis, varying input demand processes and forecasting models have been employed to obtain a comprehensive set of system responses. It is then possible to examine how these two variable conditions interact with the primary variable of interest, the information mode, to influence performance. The size of cumulative unit days of inventory and cumulative unit days of backorders have served as the primary measures of this performance. Comparison of the values obtained allows the alternate information modes to be ranked with respect to performance on these measures.

The following steps explain the sequential development of the research:

1. The characteristics of the subject two-echelon system are explained.
2. An inventory management policy is developed for this system.
3. Forecasting models for common demand patterns are selected.
4. The management policy and forecasting models are combined into an inventory control and forecasting system.
5. Programs are written to simulate this system.
6. Input demand processes are selected.
7. Simulation runs are executed to provide data for analysis.

Survey of the Literature

Forecasting in multi-echelon inventory systems has received only scant attention in the literature. Some of the studies cited in this survey do not address themselves to this problem directly but, nonetheless, present important implications for the problem. The first part of this section will discuss these direct and indirect references to forecasting. Following this, a brief description of work done in the development of optimal operating doctrines for the control of multi-echelon systems will be presented. This latter section seems appropriate since a management policy for the two-echelon system is developed in the course of this research.

Forecasting in Multi-Echelon Systems

Forrester in his book, *Industrial Dynamics* (7), was one of the earliest to focus attention on the delay and distortion factors that bear on the response of multi-stage systems. (In the general usage, multi-stage implies a system with multiple levels but with only a single unit at each of these levels; multi-echelon, on the other hand, implies the existence of more than one unit or facility at some levels; thus, a multi-stage system is a simplified version of a multi-echelon system.) In his dynamic model of a three-stage production, distribution, and retail system, he analyzes the impact of simple sales inputs on the inventory levels and production schedules. Forrester shows that, even in the presence of a noiseless, steady input, erratic behavior results. He identifies the inclusion of the distribution stage, with its distorted view of the external demand process, as the major cause of the undesirable response. Both amplification and delay factors are introduced as the distribution stage bases its replenishment policy on the orders received from the retailer. In the basic model, Forrester incorporates "no explicit forecasting procedures" (7, p.437); however, he does smooth the input data using first-order exponential smoothing at each stage to remove the noise. In an appendix, he illustrates some of the pitfalls present when implicit forecasting is used. These difficulties arise generally when a poor choice of forecasting model is made.

Strassman (24), in discussing forecasting in a complex, multi-stage system, emphasizes that forecasts should be made at a level low enough to accurately portray the real forces acting upon the system.

Magee and Boodman (15) address the problem of determining replenishment procedures in multi-stage production and distribution systems without directly discussing the use of forecasts, but the implication of these ideas for forecasting seem clear. They state that, when replenishment of stock is based on orders received from lower stages, orders become increasingly larger and less frequent, with a corresponding increase in inventory carrying costs. They offer as an alternative replenishment policy to base orders at all stages on the real demand experienced at the retail stage. This requires that demand data be transmitted to all stages in the system. In a model of a three-stage system (production, distribution, and retail), actual replenishment at the lowest stage is initiated not by the facility itself but by its supplying warehouse. The procedure impacts on production by providing more stability in the size of production orders. This has a cost-savings effect since production costs are closely related to the size of changes in production operating levels.

Meslin (16) presents ideas similar to those of Magee and Boodman. In examining the multi-echelon problem, he suggests breaking the pertinent replenishment decision into two components. Thus, a separate decision is made about the total amount of stock required and the distribution of this stock within the system. A forecast of system requirements is necessary, and two alternatives exist as to how system consumption can be measured. Basing this consumption on the movement of stock from the central warehouse to the branch facilities leads to the difficulties explained by Forrester. This movement is as much a result of the inventory management policy in effect, as it is the customer demand. The alternative is to base system consumption on the actual demand experienced at the lowest echelon. Other questions arise when this is done. Meslin considers whether the separate data streams at the first-echelon branches should be combined to form the demand for forecasting at higher echelons, or whether first-echelon forecasts should be used instead. He can give no definitive answer; however, he does suggest that different underlying demand processes will lead to different answers.

R. G. Brown, a recognized authority in the use of exponential smoothing as a forecasting method, generally has been silent about applications to multi-echelon systems. In his book, *Decision Rules for Inventory Management* (3), he discusses forecasting methods for installations at various levels of what is essentially a multi-echelon inventory system. Without elaborating on the point, he recommends that actual demand data be used at all levels in formulating forecasts.

To summarize this literature on forecasting in multi-echelon systems, it appears broadly accepted that using order quantities as a basis for forecasting at higher echelons leads to poor system performance. All of these authors either state or imply that use of demand (sales) data as a basis for these forecasts will lead to better results. Only Meslin suggests the use of lower echelon forecasts as the input information for forecasting at higher echelons. There has been little testing of these ideas.

Optimal Operating Doctrines in Multi-Echelon Systems

Clark and Scarf (5) provide the earliest statement of an optimal multi-echelon inventory policy, although the basis for this work was developed earlier by Clark (4). They prove the optimality of a (R,r) policy over n periods in the multi-stage problem. The solution procedure allows the computation of the optimal values, R and r, independently at each stage with the following primary assumption: holding and shortage costs at any level are assumed to be functions not only of the stock at that level but also of the stock at or in transit to all lower levels. Dynamic programming is used to solve for the optimal values. The solution procedure breaks down in the multi-echelon case because of an inability to deal with two questions. First, how is the optimal solution affected by allowing transshipments between facilities within an echelon? Second, how does one allocate an amount of stock that is insufficient to meet the total demands from the facilities at the next lower echelon?

Shakum (22), using an EOQ model, proves the optimality of system management over independent management in a two-echelon system with k facilities serviced by a central warehouse. The optimal order quantity, Q, is determined through the solution of a cost function. Optimal sub-allocation to the k facilities is not determined.

Two studies present solution algorithms for the two-echelon problem under the assumption of unlimited supply available at the central warehouse. Ingelhart (13) developed a periodic review model for the allocation of a quantity R (fixed over all future periods) to two satellite facilities. Skeith (23) developed a model of the two-echelon system with k branch facilities, allowing transshipments between facilities. Assuming zero internal replenishment lead times between the distribution center and the branches, he developed a total cost equation which could be solved using a nonlinear programming algorithm.

The work of Clark and Scarf, referenced earlier, was extended by Hochstaedter (11) to the multi-echelon case. An approximate optimal solution using a (R,r) policy is obtained with upper and lower bounds calculated on the values of R and r.

Summary

In this chapter, the problem in multi-echelon forecasting has been established in part as one of selecting the best mode of information on which to base higher echelon forecasts. Three modes of input information have been suggested in the literature, with a limited discussion of expected results. This research is conducted to obtain simulation test results on the performance of a two-echelon inventory system when three alternate forecasting procedures (input information modes) are used at the second echelon. The question of whether this performance varies with system input demand processes or forecast models is explored.

CHAPTER II

DEVELOPMENT OF THE FORECASTING-INVENTORY CONTROL SYSTEM

This chapter contains the development of details of the forecasting-inventory control system to be used in studying the alternate forecasting policies identified in Chapter I. Initially, the configuration of the multi-echelon system will be described. Then, the inventory management policies at each facility will be developed, with explicit consideration of the relationships between inventory policy variables and parameters which must be forecasted. Finally, forecasting models for the various demand processes to be used in the analysis will be presented.

The Material-Flow System

A simple two-echelon system, with two parallel facilities at the first level (called "branch" facilities) and one second-level facility (called the "central warehouse"), was used as the structure of the material-flow system. (See Figure 1.) The two branch facilities experience direct customer demand and the demand processes are not necessarily alike. The central warehouse has a demand composed exclusively of replenishment orders from the branches. The central warehouse orders from an external supplier. Only one commodity is assumed. Lead times for filling orders are known and backlogging is permitted at all facilities.

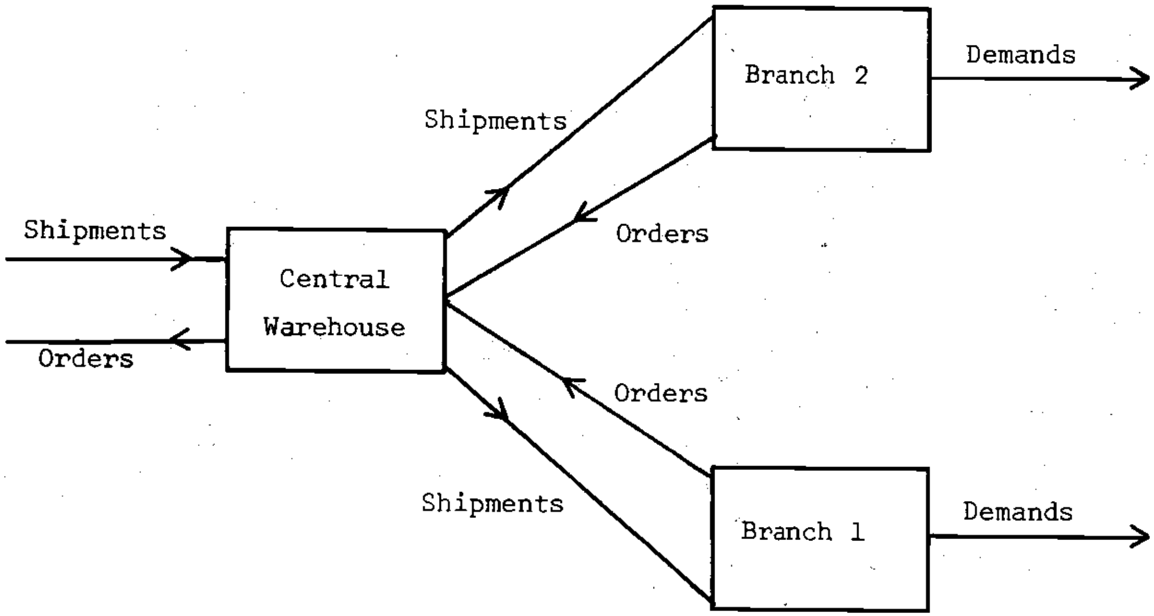

Figure 1. The Material-Flow System

The Inventory Management Policy

None of the multi-echelon or multi-stage management policies cited in Chapter I appears to possess the characteristics of completeness and simplicity essential to their being accepted on a broad scale in practice. Managers, in fact, seem perfectly willing to trade optimal or near optimal doctrines for good policies that are quick, easy to apply, and administratively inexpensive (3,20). In current inventory control practice, the well-known economic order quantity (EOQ) formula has found wide acceptance. In its most common formulation, the order quantity, Q, is expressed as

$$Q = \left[\frac{2\lambda A}{DC}\right]^{\frac{1}{2}}$$

where λ is the demand rate, A is the fixed ordering cost, D is the inventory carrying cost rate per unit time, and C is the unit variable cost of an item. While the EOQ formula provides optimal decisions only under a set of very restrictive assumptions, it gives good solutions in many applicatins (3,8). The EOQ formula will serve as the basic component of the inventory management policy in this research.

More formally, it is assumed that the inventory policy at each facility is an (R,r) policy, with R being a target inventory level and r being the reorder point. A continuous review system is conceptualized at the first echelon, but in the simulation programs (to be described later) there is only one ordering opportunity each day, so that actually the system at this level is operated as a periodic review system with a review period of one day. At the central warehouse, a transactions reporting system is used, where a review is conducted at each transaction (receipt of a request for resupply from a branch facility). The parameters, R and r, are determined by approximate means involving the simple EOQ formula and easy-to-obtain measures of the uncertainty in lead time demand. The idea is to have inventory rules considered to be good in practice. Detailed formulas are given in the next section.

In most of the literature on multi-echelon systems, optimal results are predicated on system-wide rather than independent control of the ordering decisions; however, independent management, in which each facility seeks a good or best stock control policy without concern for the system as a whole, would appear to describe what is most often found in practice. For this reason independent control is assumed in the inventory policy which follows.

Formulation of the (R,r) Policy

Let the demand process acting on the three facilities be

$$x_{it} = \mu_{it} + \epsilon_{it} \qquad i=1,2,3$$

where x_{it} = demand at facility i in period t (the central warehouse is facility 3).

μ_{it} = $E(x_{it})$, the expected demand in period t.

ϵ_{it} = the random part of the demand process, considered to be normally distributed with mean zero and variance σ_i^2, where

σ_i^2 = variance of the demand at facility i.

Note that the demand processes acting on the branches are externally imposed, while the demand process for the central warehouse is a function of the inventory policies used at the branches.

For facility i, let

I_{it} = ending inventory position on day t.

Y_{it} = on-hand inventory on day t.

B_{it} = backorder position on day t.

O_{it} = on-order position on day t.

d_{it} = demand on day t.

Then

$$I_{it} = Y_{it} - B_{it} + O_{it}$$

and

$$I_{it} = I_{i,t-1} - d_{it} + (O_{it} - O_{i,t-1})$$

Define R_i as the target inventory and r_i as the reorder point for facility i. The inventory position is reviewed and the ordering rule is

$$\begin{cases} \text{if } I_{it} \le r_i, & \text{order } R_i - I_{it}; \\ \text{if } I_{it} > r_i, & \text{do not order.} \end{cases}$$

The target inventory is defined by

$$R_i = Q_i + r_i,$$

where the quantity Q_i is computed from the EOQ formula as

$$Q_i = \left[\frac{2\hat{\lambda}_i A_i}{DC}\right]^{\frac{1}{2}}$$

$\hat{\lambda}_i$ is an estimate of the demand rate at facility i.

The reorder point is determined to give a specified degree of protection against stockouts during the replenishment lead time. The procedure used herein is

$$r_i = \tau\hat{\lambda}_i + K\hat{\sigma}_i$$

where τ is the lead time and K is a factor which depends upon the desired risk of running out of stock, the length of the lead time, and the nature of the technique used to estimate μ_i and σ_i^2.

Assumptions in the Policy Development

The following is a summary of the assumptions made in the development of the inventory management policy.

1. Independent control will be exercised at each facility.
2. Customer demand may not be made directly on the central warehouse but must come through a branch facility.
3. Backlogging is permitted.
4. Replenishment lead times are fixed.
5. Ordering costs are linear.

6. The random part of the demand process is normally distributed with zero mean and unknown variance.
7. Transshipments of stock between the two first-echelon facilities are not allowed.
8. This distribution system operates without regard for the impact of its actions on production. No restrictions are placed on order quantities or ordering frequency. No quantity discounts are available from the system's supplier.

Disallowing transshipments is a simplifying assumption which corresponds closely with actual practice, where seldom will transshipments result in a cost savings (5). Separation of the two-echelon inventory system from its production source does not prevent an examination of the impact of alternate second-echelon forecasting procedures on production.

Selection of Forecasting Models

Strassman (24) notes that, of the available methods of forecasting, only exponential smoothing of historical data offers the two-fold advantage of easy and inexpensive use. Exponential smoothing models have been developed to forecast a wide variety of demand patterns. Those of R. G. Brown (2) cover polynomial demand functions through the third order and a number of sinusoidal patterns. Winters (25) formulated a model for the constant demand process and then expanded it to include ratio seasonal effects and additive trend effects. Box and Jenkins (1) developed a general polynomial model. The polynomial models of Brown have been shown to be a special case of the Box and Jenkins model (9). Holt, et al. (12), produced a pioneering work in production and inventory control with forecasting, but their forecasting model is little different from that of Winters. Winters' work was extended by Pegels (18) to include all possible combinations of additive or ratio seasonal and trend effects.

The higher order polynomial models are not included in this study since they have been found to be of little practical use (3,10). The constant and linear (additive) trend models of Brown have fared well in tests with those of the other authors cited (9,19) and are used in this simulation study. In these same tests, Brown's sinusoidal models failed to forecast seasonal demand patterns as well as the model of Winters. Thus, Winters' basic model with additive trend and ratio seasonals is also used in this study.

Mathematical Statement of Selected Models

In the three forecasting models to follow, the quantities u, v, and w are smoothing constants. The subscript, t, refers to the current period just ended and is the time at which the forecast is generated; t' indicates that the forecast is of demand in period t + t'. The variable x_t is the actual demand experienced between times t - 1 and t.

1. *Constant Model (Single Smoothing)*

Model: $\mu_t = a$, a constant

Smoothed statistic: $S_t = ux_t + (1-u)S_{t-1}$

Estimate of coefficient: $\hat{a} = S_t$

Forecast t' periods into the future:

$$\hat{x}_{tt'} = \hat{a} \tag{2-1}$$

2. *Linear Trend Model (Double Smoothing)*

Model: $\mu_t = a + bt$

Smoothed statistics: $S_t = ux_t + (1-u)S_{t-1}$

$$S_t^{(2)} = uS_t + (1-u)S_{t-1}^{(2)}$$

Estimate of coefficients:

$$\hat{a} = 2S_t - S_t^{(2)} \tag{2-2}$$

$$b = \frac{u}{1-u}(S_t - S_t^{(2)}) \tag{2-3}$$

Forecast t' periods into the future:

$$\hat{x}_{tt'} = \hat{a} + t'\hat{b} \tag{2-4}$$

3. *Seasonal Model (Additive Trend and Ratio Seasonals)*

L is the length of a season.

Model: $\mu_t = (a+bt)c_t$

Smoothed statistic: $S_t = u\dfrac{x_t}{F_{t-L}} + (1-u)(S_{t-1}+R_{t-1})$

Seasonal factor: $F_t = v\dfrac{x_t}{S_t} + (1-v)F_{t-L}$

Trend factor: $R_t = w(S_t - S_{t-1}) + (1-w)R_{t-1}$

Forecast t' periods into the future:

$$\hat{x}_{tt'} = (S_t + t'R_t)F_{t-L+t'} \tag{2-5}$$

Forecast Control

A forecasting system must be able to recognize and signal a change in the underlying demand process so that corrective action can be taken. A change of forecast model may be necessary if the old model can no longer follow the developing historical data. In other cases, the old model may still be appropriate but at a different level of operation. Since the scope of this study is limited to an examination of the steady-state behavior of the system, a method of forecast control is not essential but is included for completeness. The method of Brown (2) is used. Two values of the smoothing constant are specified. A lower value is provided for normal operation. The higher value comes into use when the tracking signal indicates that a change has occurred in the demand process. This higher value provides more rapid discounting of past data and, therefore, allows the forecast to "home" more rapidly on the new level of operation. Brown defines the tracking signal as

$$TS = \frac{E_t}{\hat{MAD}_t}$$

where E_t, the sum of forecast errors, is

$$E_t = e_1 + e_2 + \ldots + e_t$$

and

$$e_j = x_j - \hat{x}_{j-1,1} \qquad j=1,2,\ldots,t$$

The estimate of the mean absolute deviation, $\hat{MAD}_t$, is obtained by single smoothing of the forecast errors.

$$\hat{MAD}_t = u|e_t| + (1-u)\hat{MAD}_{t-1}$$

Brown then shows that the standard deviation of the sum of forecast errors is proportional to the mean absolute deviation. The approximate relationship is

$$2\hat{\sigma}_E = \left[\frac{\pi(2-u)}{1-(1-u)^{2n}}\right]^{\frac{1}{2}} MAD$$

where n is the number of degrees of freedom in the forecast model. Assuming E is normally distributed, there is about a 5 per cent chance $|E|$ will exceed $2\sigma_E$, if the forecasting model is unbiased, that is, if the expected value of the forecast error is zero. Therefore, if $|E|$ exceeds $2\hat{\sigma}_E$, one could say with about 95 per cent confidence that this "outlier" was not caused by randomness. Brown builds in an additional safety factor by recommending that no corrective action be taken unless two successive outliers are encountered. The rule is then, if

$$TS = \frac{|E_t|}{\hat{MAD}_t} > \left[\frac{\pi(2-u)}{1-(1-u)^{2n}}\right]^{\frac{1}{2}} \qquad (2\text{-}6)$$

use the higher value of the smoothing constant.

Interface Between the Inventory Policy and the Forecast

Both the reorder point, r, and the quantity, Q, are functions of the forecast. Let the forecast be computed every T units of time. If a day is the basic unit of time in use (this will actually be the case in the simulations), then the forecasted demand rate will be expressed in units of items per T days. The replenishment lead time must be expressed as a number (not necessarily an integer) of forecast intervals. Two forecasted values are needed, the demand rate one interval into the future and the expected lead time demand. When the constant model is used, one forecast is a real valued multiple of the other; with the trend or seasonal models the relationship is more intricate. As an example, the trend model is examined.

Let $\hat{x}_{t1}$ be the forecasted demand rate one forecast interval into the future and $\hat{x}_{tm}$ be the demand rate m forecast intervals into the future, where $m = \tau/T$. Then, from Equation 2-4,

$$\hat{x}_{t1} = \hat{a} + \hat{b}, \tag{2-7}$$

and

$$\hat{x}_{tm} = \hat{a} + m\hat{b}$$

Now let n be the largest integer less than or equal to m and let s be the fractional difference between them, i.e. $s = m - n$; then the expected demand over a lead time, $\hat{H}$, during the next lead time, will be

$$\hat{H} = \left\{ \sum_{j=1}^{n} (\hat{a}+j\hat{b}) \right\} + s\{\hat{a} + \hat{b}(n+1)\} \tag{2-8}$$

For the seasonal model, $\hat{H}$ can be computed in a similar way from Equation 2-5 to yield

$$\hat{H} = \left\{ \sum_{j=1}^{n} (S_t+jR_t)F_{t-L+j} \right\} + s\{(S_t+jR_t)F_{t-L+n+1}\} \tag{2-9}$$

The quantity $\hat{H}$ will then be interpreted as the demand rate, λ, for use in the EOQ and reorder point formulas. The latter formula can now be revised as

$$Q = \left[\frac{2\hat{H}A}{DC}\right]^{\frac{1}{2}} \tag{2-10}$$

The carrying cost rate, D, is then based on a unit of time being the length of a lead time.

In the reorder point formula, the safety stock is expressed in terms of the standard deviation of the random noise. However, this value is unknown. Brown (2) shows that, if the noise in the demand process is normally distributed with zero mean and the noise samples are serially independent, then the forecast errors will also be normally distributed with zero mean. He also shows that σ_e is approximately equal to 1.25 times the mean absolute deviation of forecast errors. Therefore, the reorder point formula can now be written as

$$r = \hat{H} + k(1.25)\hat{MAD}_t\left(\frac{\tau}{T}\right) \tag{2-11}$$

where k is a safety factor obtained from tables of the standard normal distribution. The above is an approximate result. It is based on the assumption, previously mentioned, of serial independence of noise samples. Also, the simple multiplicative method of extending the mean absolute deviation over a lead time assumes that $\hat{MAD}$ is a linear function of the input noise, which is not true (2). The approximate expression should, however, yield satisfactory results in this research.

Summary

In this chapter, mathematical expressions have been presented which describe the integrated operation of forecasting and inventory control for the two-echelon inventory system which will be tested. An inventory management policy was developed. Three forecasting models were selected for use in these tests, and a method of coupling these models with the management policy was described.

CHAPTER III

TEST PROCEDURES

In this chapter procedures will be established to guide the testing of the forecasting-inventory control system that was developed in Chapter II. The general test design will be presented, followed by a discussion of statistics selected for output from the test runs. Then input demand processes will be selected, values will be assigned to parameters of the forecast and inventory control equations, and initial values will be assigned to model variables. Finally, considerations in the selection of simulation run time will be presented. Detailed discussion of simulation model development can be found in Appendix A. References in this chapter to characteristics of these models will be restricted to those necessary for a clear presentation of test procedures.

General

The research objective is to evaluate the three alternate ways of defining input information for forecasting at higher echelons. This is accomplished by testing these three forecasting procedures in a two-echelon inventory system with the three selected forecasting models and various demand processes, yet to be described.

The second-echelon forecasting procedures are restated as:

1. Inputs for forecasting at the second echelon are the order quantities requested by the first-echelon branches.

2. Inputs for forecasting at the second echelon are the demands made by customers on the first-echelon branches.

3. Inputs for forecasting at the second echelon are the forecasts made by the first-echelon branches.

Hereafter, the above alternate information input modes are referred to as Procedures 1, 2, and 3, respectively. Implementation of the first and second procedures requires no explanation; however, use of the third procedure is not so obvious. In this research, the forecast for the second echelon with this procedure is obtained by simple addition of branch facility forecasts made over the lead time of the central warehouse. The exact details of these computations are explained in Appendix A.

Nine simulation models have been constructed to reflect all combinations of forecasting models and second-echelon forecasting procedures. Each selected demand process is imposed on each of these nine models and system response is noted. The desired result is to be able to make some statement of the comparative performance of second-echelon forecasting procedures with these demand processes. Effect of the forecasting model in use is also noted.

Criteria for Evaluation and Output Statistics

The following two criteria have been used to judge the performance of the three alternate forecasting procedures:

1. Size of cumulative unit days of inventory.

2. Size of cumulative unit days of backorders.

Each of these criteria can be directly related to an inventory cost; thus, the forecasting procedure which achieves lowest values for these statistics can be considered to have performed best. It is unlikely, however, that the procedure which achieves lowest inventory stocks will also result in the smallest number of backorders. No attempt has been made to combine these criteria into a single measure of performance; rather, performance has been judged against each, independently.

Other statistics that have been extracted from the simulations are central warehouse forecasts, reorder points, estimates of the mean absolute deviation, and order quantities. Also, the number of orders placed by the central warehouse has been determined. These values assist in understanding why the values of unit days of inventory and backorders are achieved. The size of orders placed by the central warehouse are of interest in determining the effect that each second-echelon forecasting procedure would have on a production facility, were one included.

Daily values of net inventory and inventory position at each facility are also printed by the simulation programs. These values, while primarily of aid in debugging and validating the simulation models, also assist in the investigation of inconsistent or unexpected results.

Selection of Demand Processes

Only a small fraction of the demand processes which one might expect to encounter in inventory systems can be studied in this research. Those that have been selected can be described as "steady sellers" with a low probability of zero demand on any day. Processes of this type are common in practice.

These processes are generated so that the demand pattern over time will fit the particular forecasting model in use, e.g., when the trend forecasting model is used, the demand process reflects a linear growth over time. Only the steady-state behavior of the system is simulated. This means that, during a particular simulation, the underlying demand process does not change, although demand may be changing as a function of time. Thus, only a single forecasting model need be used in each simulation.

The variable factors in these processes are the constant component and the noise variance. Trend and seasonal components remain fixed during all tests as follows:

1. The trend factor reflects a growth in one year of one-half the value of the constant component.

2. The seasonal factors reflect a general sinusoidal pattern with a cycle of six months and a starting amplitude of one-fourth the value of the constant component. Since the seasonal model used is multiplicative, this amplitude grows with the trend.

The values for the variable components in the demand processes were selected to reflect four broad categories, to achieve a wide range of responses. These categories with actual test values are:

1. *High variance demands at both branch facilities.*

	Branch 1	Branch 2
Constant component	16.00	24.00
Noise standard deviation	8.00	12.00

2. *Low variance demands at both branch facilities.*

	Branch 1	Branch 2
Constant component	16.00	24.00
Noise standard deviation	2.00	3.00

3. *Dissimilar demands at the two branches.*

	Branch 1	Branch 2
Constant component	6.00	24.00
Noise standard deviation	2.00	12.00

4. *Similar demands at the two branches.*

	Branch 1		Branch 2	
Constant component	24.00	6.00	24.00	6.00
Noise standard deviation	12.00	2.00	12.00	2.00

These values of customer demand are generated on a daily basis using a normal deviate generator. (For details of the process generators, see Appendix A.) Cumulative demand data over a forecast interval are sums of normally distributed random variables and, therefore, are also normally distributed. The theoretical values of the mean (constant component) and the noise variance over a forecast interval can be easily calculated.

Assignment of Parameter Values

Assignment of values to parameters in the simulation models has been done somewhat arbitrarily, but these values are felt to be realistic.

Parameters Particular to a Facility

Forecasts for both branch facilities are updated every seven days. This forecast interval has been extended to 30 days for the central warehouse. Preliminary tests have shown that use of a shorter interval at this location would badly prejudice that forecasting procedure based upon orders placed by the branches, due to frequent intervals of zero demand (orders). Replenishment lead times for the branches are set at 14 days. Lead time for the central warehouse is 30 days. Fixed ordering costs at the central warehouse will be $100.00; at both branch facilities this cost will be $50.00.

Parameters Common to All Facilities

Two levels of each smoothing constant are provided. The higher level is used only when the tracking signal indicates that the forecast is out of control. All smoothing constants will have a low level of .10 and a high level of .40. The safety factor selected for determining the amount of safety stock will be 1.2 at each facility. This value yields a theoretical probability of 78.3 per cent that no out-of-stock condition will arise at a branch facility in a replenishment lead time. Factors for forecast control have been computed by solving for the value of the right-hand side of Inequality 2-6 to yield about a 95 per cent probability that a change has occurred in the underlying demand process when the tracking signal indicates an out-of-control condition. The control values are

Constant Model - 5.60

Trend Model - 4.16

Seasonal Model - 3.56

Initial Conditions

Variables are initialized to reflect an equilibrium condition. This state of equilibrium can be visualized by imagining that each facility at time zero has perfect knowledge of the customer demand process, for this one instant. Calculation of initial values for variables in the forecast models are shown below. Once these values have been determined and the estimate of the mean absolute deviation has been initialized, Equation 2-11 can be solved for the starting value of the reorder point. The starting values for net inventory and inventory position at each facility have been arbitrarily assigned as the economic order quantity, Q, which is determined from Equation 2-10. The estimate of the mean absolute deviation is initialized using an approximate expression by Brown (2).

$$MAD = (2/\pi)^{1/2}\left(2/(2-u)\right)^{1/2}\sigma$$

In this expression σ is the standard deviation of the noise in the demand process, and u is the smoothing constant. Initial values of variables in the forecast models follow.

Constant Model. Let the mean daily demand rate be d. The initial value of the forecast over a forecast interval, T, is

$$\hat{x}_{0,1} = dT$$

The initial value of the forecast over a lead time, τ, is

$$\hat{H} = d\tau$$

Trend Model. The starting value of the constant coefficient is

$$\hat{a}_0 = dT$$

If (tr) is the slope of the trend line being generated, then the trend coefficient is

$$\hat{b}_0 = (tr)T$$

Equations 2-2 and 2-3 are then solved simultaneously to yield initial values for the statistics as

$$S_0 = \hat{a}_0 - \frac{(1-u)}{u}\hat{b}_0$$

$$S_0^{(2)} = \hat{a}_0 - \frac{(2-2u)}{u}\hat{b}_0$$

Initial values for forecasts of demand over a forecast interval and over a lead time are obtained by solving Equations 2-7 and 2-8, respectively, using the initial values of the coefficients obtained above.

Seasonal Model. The procedure is quite similar to that for the trend model.

$$S_0 = dT$$

$$R_0 = (tr)T$$

The seasonal factors can be set equal to those factors being used to generate the seasonal effect in demand. The forecast over a forecast interval is

$$\hat{x}_{0,1} = (S_0+R_0)F_{0-L+1}$$

The forecast of lead time demand is obtained by solving Equation 2-9.

Run Time Validation

Nine simulation models were constructed to include all combinations of forecast models and second-echelon forecasting procedures. Preliminary tests on these simulation models revealed little difference in system response over the first 90 days of simulation, regardless of which forecasting procedure was employed at the second-echelon. As a result of these initial runs, the first 90 days are considered simulator warm-up time in the record runs. Output data from these first 90 days are disregarded. Runs of one, two, and three years were conducted to determine what length of simulation would be necessary. Results at the end of two years were consistent with those at the end of three years. All record runs are based upon a three-year simulation beyond the 90-day warm-up period.

Summary

In this chapter, procedures to be used in testing the three alternate modes of input information were described. Criteria for evaluating performance and simulation output statistics were selected. Then, input demand processes were chosen, values were assigned to system parameters, and the method of initializing system variables was explained. Finally, considerations in the selection of simulation run time were discussed.

CHAPTER IV

RESULTS

The simulation models in Appendix A were tested with the demand processes described in Chapter III. Each second-echelon forecasting procedure was tested using each forecast model and demand process, with one exception. The lower level of constant component and noise variance for the process category "Similar Demand" was added after the start of testing. The purpose of this addition was to gain a clearer distinction between Procedures 2 and 3, and this demand process was not tested with Procedure 1.

In this chapter, the results of the 42 simulation tests will be presented and discussed. Summary data is presented in the next section. Additional simulation output statistics appear in Appendix B.

Simulation Results

The summary data reflecting comparative performance of the three forecasting procedures appear in Tables 1-15. Three different types of cost are related to the size of values in these tables. Inventory carrying costs and the costs of incurring backorders are reflected by the average daily unit days of inventory and backorders, respectively. One variable aspect of production costs can be related to the size of the standard deviation of central warehouse order quantities.

Table 1. Results with Low Variance Demands, Constant Model

Procedure	Average Daily Unit Days of Inventory	Average Daily Unit Days of Backorders	Number of Orders	Total Stock Ordered	Standard Deviation of Order Quantities
1	2169.02	.16	24	44,371	327.53
2	1809.70	.70	24	43,116	355.16
3	1737.75	.77	22	43,127	372.35

Table 2. Results with High Variance Demands, Constant Model

Procedure	Average Daily Unit Days of Inventory	Average Daily Unit Days of Backorders	Number of Orders	Total Stock Ordered	Standard Deviation of Order Quantities
1	2362.35	.16	24	44,713	339.73
2	1975.57	.49	23	42,545	347.71
3	1965.67	.36	21	41,896	374.63

Table 3. Results with Dissimilar Demands, Constant Model

Procedure	Average Daily Unit Days of Inventory	Average Daily Unit Days of Backorders	Number of Orders	Total Stock Ordered	Standard Deviation of Order Quantities
1	1831.06	.15	15	32,406	76.18
2	1523.77	.15	15	32,034	42.28
3	1566.44	.15	15	32,028	267.99

Table 4. Results with Similar Demands, High Level, Constant Model

Procedure	Average Daily Unit Days of Inventory	Average Daily Unit Days of Backorders	Number of Orders	Total Stock Ordered	Standard Deviation of Order Quantities
1	2525.88	.26	27	51,987	323.57
2	2218.98	.26	26	52,374	377.97
3	2107.72	.31	26	54,109	360.50

Table 5. Results with Similar Demands, Low Level, Constant Model

Procedure	Average Daily Unit Days of Inventory	Average Daily Unit Days of Backorders	Number of Orders	Total Stock Ordered	Standard Deviation of Order Quantities
2	1028.43	5.19	12	12,318	206.38
3	947.43	6.06	11	11,881	200.24

Table 6. Results with Low Variance Demands, Trend Model

Procedure	Average Daily Unit Days of Inventory	Average Daily Unit Days of Backorders	Number of Orders	Total Stock Ordered	Standard Deviation of Order Quantities
1	2969.04	.66	35	86,611	596.33
2	2520.87	.68	38	85,633	272.65
3	2511.51	.70	38	85,653	270.46

Table 7. Results with High Variance Demands, Trend Model

Procedure	Average Daily Unit Days of Inventory	Average Daily Unit Days of Backorders	Number of Orders	Total Stock Ordered	Standard Deviation of Order Quantities
1	2933.44	0	34	86,481	547.60
2	2657.18	.08	37	84,190	261.03
3	2589.94	.08	36	85,244	416.33

Table 8. Results with Dissimilar Demands, Trend Model

Procedure	Average Daily Unit Days of Inventory	Average Daily Unit Days of Backorders	Number of Orders	Total Stock Ordered	Standard Deviation of Order Quantities
1	2254.09	.42	24	65,562	556.83
2	2059.61	1.29	25	64,880	650.34
3	2068.72	1.76	27	64,916	581.78

Table 9. Results with Similar Demands, High Level, Trend Model

Procedure	Average Daily Unit Days of Inventory	Average Daily Unit Days of Backorders	Number of Orders	Total Stock Ordered	Standard Deviation of Order Quantities
1	3391.20	.41	39	102,523	487.84
2	2869.75	.41	41	101,626	341.81
3	2883.66	.41	39	100,272	447.98

Table 10. Results with Similar Demands, Low Level, Trend Model

Procedure	Average Daily Unit Days of Inventory	Average Daily Unit Days of Backorders	Number of Orders	Total Stock Ordered	Standard Deviation of Order Quantities
2	1662.23	.05	21	25,886	132.65
3	1658.14	.05	21	25,857	130.72

Table 11. Results with Low Variance Demands, Seasonal Model

Procedure	Average Daily Unit Days of Inventory	Average Daily Unit Days of Backorders	Number of Orders	Total Stock Ordered	Standard Deviation of Order Quantities
1	3015.27	2.20	30	86,870	576.13
2	2389.11	3.33	32	87,719	565.97
3	2570.94	5.86	33	86,290	502.16

Table 12. Results with High Variance Demands, Seasonal Model

Procedure	Average Daily Unit Days of Inventory	Average Daily Unit Days of Backorders	Number of Orders	Total Stock Ordered	Standard Deviation of Order Quantities
1	3030.25	2.86	31	87,728	666.23
2	2531.81	6.12	31	86,868	713.58
3	2607.84	11.10	31	86,826	673.81

Table 13. Results with Dissimilar Demands, Seasonal Model

Procedure	Average Daily Unit Days of Inventory	Average Daily Unit Days of Backorders	Number of Orders	Total Stock Ordered	Standard Deviation of Order Quantities
1	2502.75	1.25	27	65,073	497.72
2	2109.12	11.86	26	64,230	610.63
3	2213.99	8.20	26	63,491	513.69

Table 14. Results with Similar Demands, High Level, Seasonal Model

Procedure	Average Daily Unit Days of Inventory	Average Daily Unit Days of Backorders	Number of Orders	Total Stock Ordered	Standard Deviation of Order Quantities
1	3429.28	.81	35	101,501	619.99
2	2820.21	6.22	34	100,520	586.76
3	2741.49	14.75	35	102,906	686.34

Table 15. Results with Similar Demands, Low Level, Seasonal Model

Procedure	Average Daily Unit Days of Inventory	Average Daily Unit Days of Backorders	Number of Orders	Total Stock Ordered	Standard Deviation of Order Quantities
2	1325.43	.25	18	25,787	255.78
3	1437.08	5.26	17	26,889	411.85

The value, average daily unit days of inventory, applies to the total system (both branches and the central warehouse). The value, average daily unit days of backorders, however, includes only those backorders incurred at the first echelon. Backorders incurred at the central warehouse have no direct impact on customer service. The indirect impact is reflected in backorders at the branches. The quantities, number of orders and total stock ordered, reflect the full three years of simulation time.

Discussion of Results

The results of this research, as displayed in Tables 1-15, reveal few consistent patterns. Best performance, as measured with the three cost-related factors, varies with different demand processes and different forecast models. One general comment that can be made, at this point, concerns the use of Procedure 1 (forecasting second-echelon demand based upon first-echelon orders). This procedure consistently results in the largest value of inventory stock and the smallest number of backorders. The reason is that use of this procedure leads to poor forecasts. The management policy overcompensates for this with large safety stocks. These poor forecasts are reflected by the large values of mean absolute deviation. (See Appendix B.) In the discussion to follow, no further mention of Procedure 1 will appear until the variance of order quantities is addressed.

Size of Inventory Stocks

Best performances with respect to this factor, based upon Column 2 of Tables 1-15, are presented in capsule form in Table 16.

Table 16. Performance Summary on Size of Inventory Stocks

Forecast Model	DEMAND PROCESS				
	Low Variance	High Variance	Dissimilar	Similar High Level	Similar Low Level
Constant	Procedure 3	Little Difference	Procedure 2	Procedure 3	Procedure 3
Trend	Little Difference	Procedure 3	Little Difference	Little Difference	Little Difference
Seasonal	Procedure 2	Procedure 2	Procedure 2	Procedure 3	Procedure 2

As mentioned above, only Procedure 2 (second-echelon forecasts based upon customer demand) and Procedure 3 (second-echelon forecasts achieved by combining first-echelon forecasts) are considered. When there appears to be little difference between the performances of the two procedures, this is indicated in the table.

When the seasonal model is used, Procedure 2 gives a better performance with all demand processes, except similar demands at both branches (with higher level of the constant component and variance).

Backorders

In comparing Procedures 2 and 3 with respect to this factor, one may state that, generally, the procedure which has the lowest values of average daily unit days of inventory will incur the most backorders. Yet, the results in Tables 1-15 show exceptions. The most noteworthy seems to be that, with the seasonal model, Procedure 2 yields both low values of inventory stock and backorders.

Variance of Order Quantities

No distinct pattern is apparent in the data. While Procedure 1 performs well and all demand processes imposed on the constant model, it does poorly with the trend model, with the exception of dissimilar demands. With the seasonal model, Procedure 1 yields a better result when high variance and dissimilar demand processes are imposed. The behavior of Procedures 2 and 3 can only be described as erratic.

Number of Orders and Total Quantities Ordered

No clear relationship can be established between these values and the three cost-related factors. It is interesting to note that, frequently, the procedure which results in the lowest value of average daily unit days of inventory will also result in the largest quantity of stock being ordered. This indicates that timing of first-echelon orders on the central warehouse has an important effect.

Summary

In this chapter, results of the simulation tests have been presented. These results indicate that comparative performance between the three forecasting procedures is highly sensitive to the forecast model and input demand process. None of the procedures dominates with best performance on all three cost factors for a particular forecast model or demand process. The outcomes with Procedure 2 on the seasonal model come closest to reflecting a uniformly good performance. Procedures 2 and 3 yield similar results in many of these tests.

CHAPTER V

CONCLUSIONS AND RECOMMENDATIONS

In this chapter, conclusions drawn from the test results of the previous chapter, and recommendations for further research in multi-echelon forecasting, will be presented.

Conclusions

The conclusions to follow are restricted by the narrow scope of this research. Before reviewing these limitations, the three second-echelon forecasting procedures will be restated once more for reference.

1. Forecasts at the second echelon are based upon orders placed by the first-echelon branches.

2. Forecasts at the second echelon are based upon the customer demand experienced by the first-echelon branches.

3. Forecasts at the second echelon are based upon the forecasts made by the first-echelon branches.

Only one particular type of demand process, the "steady seller," was used in evaluating the above forecasting procedures. Within this broad classification of processes, only five examples have been examined. The system studied has only two echelons and two branch facilities at the first echelon. The three forecasting models used, while representative of those found in practice, are not exclusive.

In light of the above qualifications, the conclusions are:

1. Comparative performance of the three forecasting procedures with respect to inventory carrying costs, backorder costs, and impact on production is highly sensitive to the input demand process and the forecasting model in use.

2. That forecasting procedure which leads to smallest inventory stocks will generally produce the largest number of backorders.

3. With respect to inventory carrying and backorder costs, the performance of Procedure 1 is inferior to that of Procedures 2 and 3. Although use of Procedure 1 yields the lowest values of unit days of backorders, the good result on this factor is offset by the poor showing on unit days of inventory. Use of Procedure 1 yields a result on this latter factor which is, on the average, 19½ per cent higher than the best result obtained with Procedure 2 or 3.

4. Use of Procedure 2 with the seasonal model yields consistently good results. No other preference can be stated between Procedures 2 and 3 for a particular demand process or for a particular forecasting model. Substitution of one procedure for the other in any of the tests results in no more than an 8 per cent reduction in unit days of inventory; and, except with the seasonal model, values of unit days of backorders for the two are quite close.

5. Procedure 3, which is not widely recognized, provides a good alternative to Procedure 2, which is well known.

Recommendations

Possible extensions of this research are numerous.

A more exhaustive testing with steady-type input demand should give a clearer indication of how the two-echelon system responds to this type of process. Other common types of input process, such as lumpy, sporadic demand, can be tested with these models, although a change in process generators may be necessary. Such an adjustment in the simulation models is easily accomplished.

It would be of interest to observe the changes that occur in the comparative performance of the three second-echelon forecasting procedures, when independent control is replaced by a form of system management. This might be accomplished by the following steps:

1. Central warehouse inventory position could be redefined as the sum of all stock on hand or on order at this location and at the branches, and all stock in transit to the branches.

2. The central warehouse reorder point would then be based on a system lead time, which would be the sum of its own replenishment lead time and the longest of the branch lead times.

3. Branch facilities could be allowed to continue their independent ordering policies.

Expansion of the simulation models in this research to include additional branch facilities should produce different results, especially when Procedure 1 is used, since the pattern of orders to the central warehouse could develop a smoother appearance.

In this research, only the steady-state behavior of the system has been considered. A natural extension of these tests would be to observe the performance of the forecasting procedures when simple changes in the demand process occur. Pulse, step, or ramp inputs could be easily simulated to represent these changes.

The second-echelon forecasting interval was selected in Chapter III to decrease the variability of forecasts made with Procedure 1. Influence of this interval length on Procedures 2 and 3 was not considered. A replication of these experiments using a shorter second-echelon forecasting interval would be of interest in clarifying the effect of this parameter.

APPENDIX A

SIMULATION MODELS

From a standpoint of simulation design, the key independent variables subject to change are forecasting models and second-echelon forecasting procedures. Since there are three of each, nine different computer simulation models will be necessary. These nine, however, are simply variations of a single basic model. Initially, the description in this appendix will present the broad considerations in the design of this basic model. A macro flow diagram of this basic simulator is shown in Figure 2. Following this, the details of the nine individual models will be discussed under a separate subheading. Later sections will relate steps taken to validate the models and limitations of these simulators.

General Description

The simulations are written in the FORTRAN IV language. This language is not only easy to use but also very efficient in execution. The basic model consists of a main program and seven or eight (the number is dependent on the forecasting procedure in use) separate subprograms. Schmidt and Taylor (21) have produced a well-designed simulator of a single-stage inventory system, and their general approach has been followed in developing the basic model for this research.

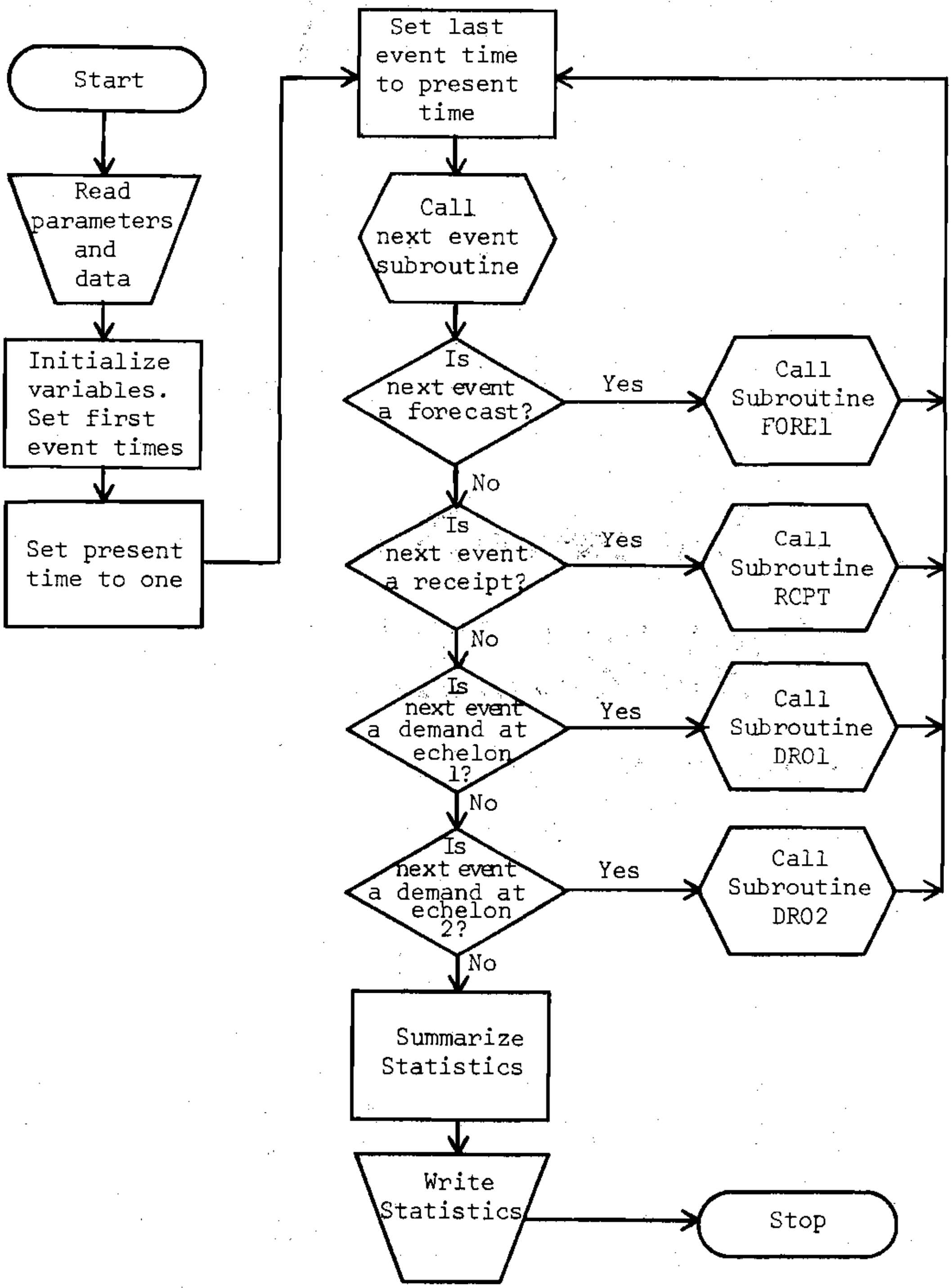

Figure 2. Macro Flow Diagram of Basic Simulator

The next-event concept of progressing the simulation through time is used. In this method a matrix of all possible events is maintained with a next time for execution specified for each. After an event is executed, its time value in the matrix is incremented. A search of the matrix is then conducted to locate that event with the next earliest execution time. This event is selected and a master clock is advanced to its execution time. Execution follows and this sequence continues until a pre-specified termination time is reached. The events that have been identified for simulation in the two-echelon inventory system are:

1. Forecast for facility i.
2. Receipt of goods at facility i.
3. Demand experienced at facility i (at the central warehouse, facility three, this demand would be in the form of an order from a branch facility).
4. Review conducted at facility i.
5. Order placed at facility i.
6. Receipt scheduled for facility i.

In addition, two administrative events are included for control purposes. The first of these prints statistics at the end of the simulation warm-up period. The second is an event which terminates the simulation.

While establishing the event structure of the model, it becomes evident that certain decisions must be made which further specify particular characteristics of the system. One question concerns priorities in execution of events. The basic unit of time in the simulations is a day, and on a given day any of the above-listed events may be scheduled. Which will be executed first? The sequence of events as given in the preceding paragraph would appear logical. We will assume that the record of stock on hand is updated only at the end of the work day. The new value of inventory position is then compared with the reorder point and an order placed, if required. If the demand is experienced at the central warehouse and stock is available to fill the order, the receipt at the branch facility is scheduled. If the central warehouse orders, it schedules its own receipt since the next higher echelon is not being simulated. Reviews, order placements, and receipt schedulings are viewed here as a natural consequence of a demand being registered; therefore, of these four events only demands must be included in the event matrix. Forecasts and receipts would have occurred earlier in the day. It makes no difference which of these two events comes first.

Priorities between facilities must also be specified. An arbitrary decision has been made to execute a particular event first for Facility 1 (Branch 1) and last for Facility 3 (the central warehouse). These priorities should have little effect on the outcomes of interest.

Two other decisions have been made in design to add more realism to the simulations. If the central warehouse experiences demands on the same day from both branch facilities and insufficient stock is on hand to completely fill both orders, then each branch will be shipped a share of the available stock proportionate to the size of its order.

Also, a one-day delay between placement of an order by a branch facility and arrival of this request at the central warehouse has been built into the model.

Individual Simulation Models

In this section, the individual simulation models will be described and illustrated. The base simulation program in this series of nine uses a constant forecast model and uses first-echelon orders as the input information for second-echelon forecasting. This model was viewed as the simplest to construct, and all other simulators were developed by adding to and deleting from this base model.

The most extensive changes occur in these simulators as the forecasting model is changed; therefore, simulation of the system with separate forecasting models forms the organizational subdivisions of this section.

Base Simulator (Constant Model, Procedure 1)

This model consists of a main program and seven subprograms. The purpose of the main program is to input data, move the simulation through time, store accumulated statistics in arrays, and print output. The subprograms are listed below with a brief description of their function.

1. *Subroutine NEXT.* This subroutine determines which event will occur next in the simulation.

2. *Subroutine UPDATE.* This subroutine accumulates both unit days of inventory and unit days of backorders.

3. *Subroutine FORE1.* This subroutine computes forecasts, updates the estimate of the mean absolute deviation, and computes a new reorder point.

4. *Subroutine RCPT.* This subroutine receives scheduled arrivals of shipments at a facility.

5. *Subroutine DRO1.* This subroutine causes a demand to be generated daily at first-echelon facilities. It then executes a review of inventory position and orders from the central warehouse, if necessary.

6. *Subroutine XNORM.* This subroutine generates the normally-distributed demand. It is called from Subroutine DRO1.

7. *Subroutine DRO2.* This subroutine accepts orders at the central warehouse from the branches, ships stock to and schedules arrivals for the branches, reviews central warehouse inventory position, orders for the central warehouse, if necessary, and schedules arrivals of orders at the central warehouse.

In the illustrations of these subroutines (Figures 3-9), only variables particular to each subroutine will be defined. A complete listing of general variables with definitions will be found in the main program (Figure 10).

Event Scheduling. Although the next-event concept is used, a next-event matrix is not employed as such. Rather, 12 individual events are arranged sequentially as follows:

Event 1 - Statistical printout after simulator warm-up.

Event 2 - Termination of simulation.

Event 3 - Forecast at Facility 1.

Event 4 - Forecast at Facility 2.

Event 5 - Forecast at Facility 3.

Event 6 - Receipt of goods at Facility 1.

Event 7 - Receipt of goods at Facility 2.

Event 8 - Receipt of goods at Facility 3.

Event 9 - Demand at Facility 1.

Event 10 - Demand at Facility 2.

Event 11 - Order accepted from Facility 1 by Facility 3.

Event 12 - Order accepted from Facility 2 by Facility 3.

The next execution time for each event is stored in this same sequence in a vector, EM. After an event is executed, Subroutine NEXT (Figure 3) looks at this vector and finds the smallest time value.

```
      SUBROUTINE NEXT(NEVENT,TIME)
C     PURPOSE - TO SELECT THE NEXT EVENT.
C     AMIN IS USED AS A TEMPORARY STORAGE
C     LOCATION FOR EVENT TIMES.
      DIMENSIONS EM(12),AIQ(3),BACK(3),PIL(3)
      COMMON PIL,EM,DUM(6),AIQ,BACK,DUMMY(2),TLAS
      INTEGER EM,TIME,TLAS
      INTEGER AIQ,BACK,PIL,AMIN
      AMIN=EM(1)
      NEVENT=1
      DO 10 I=2,12
      IF(AMIN.LE.EM(I)) GO TO 10
      AMIN=EM(I)
      NEVENT=I
   10 CONTINUE
      TIME=AMIN
      IF(TIME.EQ.TLAS) GO TO 11
      CALL UPDATE(AIQ,BACK)
   11 RETURN
      END
```

Figure 3. Subroutine NEXT, Base Simulator

It then places this value in the variable TIME and places the sequence number of this time value in the variable NEVENT. Back in the main program, this sequence number in NEVENT controls the routing of the simulation to the proper event subroutine. This routing is accomplished with a computed GO TO statement. The two FORTRAN statements in the main program which accomplish these actions are:

```
      CALL NEXT (NEVENT, TIME)
                 :
      GO TO (220,280,230,230,230,250,250,250,260,260,270,
             270),NEVENT
```

The sequence number in NEVENT determines which labelled event will be executed.

The time values in vector EM are updated in the event subroutines.

<u>Updating Unit Days of Backorders and Inventory</u>. At least two events occur each day, since first-echelon demands are generated each day. In Subroutine NEXT, after the next event has been selected, its execution time (in variable TIME) is compared to the time of the last executed event (in variable TLAS). If these two time values are different, then Subroutine UPDATE (Figure 4) is called to revise cumulative unit days of inventory and backorders in vectors AIQ and BACK, respectively. In this subroutine, the net inventory for each facility in vector PIL is examined and added to AIQ, if positive or zero, and to BACK, if negative. These vectors have three components, one for each facility. Thus, only once each day, at the end of the day, are these statistics revised.

```
      SUBROUTINE UPDATE(AIQ,BACK)
C     PURPOSE - TO UPDATE THE CUMULATIVE UNIT DAYS OF
C     INVENTORY AND BACKORDERS.
      DIMENSION PIL(3),BACK(3),AIQ(3)
      COMMON PIL,DUMM(24),TIME,NEVENT,TLAS
      INTEGER TIME,TLAS
      INTEGER AIQ,BACK,PIL
      JJJ=TIME-TLAS
      DO 21 I=1,3
      IF(PIL(I).GE.0) GO TO 20
      BACK(I)=BACK(I)+PIL(I)*JJJ
      GO TO 21
   20 AIQ(I)=AIQ(I)+PIL(I)*JJJ
   21 CONTINUE
      RETURN
      END
```

Figure 4. Subroutine UPDATE, Base Simulator

Forecasting. The forecasting subroutine, FORE1, is illustrated in Figure 5. The forecast of the demand over a forecast interval is placed in vector SM. In the constant model, no variable exists for the value of the forecast over a lead time. Whenever this value is needed, it is expressed as multiple of SM. The input information for forecasting is contained in the vector SDF. For Facilities 1 and 2, this value is the sum of demands since the last forecast; for Facility 3, the value is the sum of orders since the last forecast. After the forecast is computed, the reorder point is revised and placed in vector RP.

Receiving Shipments. Arriving shipments are contained in vector RCQ. Since the central warehouse may have insufficient stock to completely fill an order from a branch, the vector QR is provided to allow shipments to be split. Subroutine RCPT (Figure 6) handles receipts for all facilities. The quantity in RCQ(K), where K is the facility receiving the shipment, is added to the net inventory, PIL(K).

```
      SUBROUTINE FORE1(ERT,SM,SMAD,SDF,IND,RP,EM)
C     PURPOSE - TO UPDATE FORECASTS.
      DIMENSION SDF(3),SM(3),ERT(3),SMAD(3),ALO(3),AHI(3),CNTR(3),IND(3)
     A,RP(3),TAU(3),DELT(3),EM(12)
      COMMON DUMM(27),TIME,NEVENT,TLAS,ALO,AHI,CNTR,TAU,DELT,DU(13),XXX
      INTEGER EM,TAU,TIME,DELT
      INTEGER SDF,RP
      K=NEVENT-2
C     SUM IS A REAL VARIABLE EQUIVALENT TO THE INTEGER SDF(K).
      SUM=SDF(K)
C     ERR IS THE FORECAST ERROR.
      ERR=SUM-SM(K)
C     THE FOLLOWING 3 STATEMENTS REVISE THE SUM OF ERRORS AND THE
C     ESTIMATE OF MAD, AND COMPUTE THE TRACKING SIGNAL.
      ERT(K)=ERT(K)+ERR
      SMAD(K)=ALO(K)*ABS(ERR)+(1-ALO(K))*SMAD(K)
      TS=ABS(ERT(K))/SMAD(K)
      IF(TS.GT.CNTR(K)) GO TO 30
      IND(K)=0
      GO TO 31
   30 IND(K)=IND(K)+1
      IF(IND(K).LE.1) GO TO 31
      ERT(K)=0
C     THE FOLLOWING STATEMENT COMPUTES THE FORECAST
C     WITH HIGH LEVEL OF SMOOTHING CONSTANT.
      SM(K)=AHI(K)*SUM+(1-AHI(K))*SM(K)
      GO TO 32
C     THE FOLLOWING STATEMENT COMPUTES THE FORECAST
C     WITH LOW LEVEL OF SMOOTHING CONSTANT.
   31 SM(K)=ALO(K)*SUM+(1-ALO(K))*SM(K)
   32 CONTINUE
C     UAT AND TLED ARE REAL EQUIVALENTS OF THE INTEGERS TAU(K) AND
C     DELT(K).
      UAT=TAU(K)
      TLED=DELT(K)
      T=UAT/TLED
      XYZ=SM(K)*T+XXX*1.25*SMAD(K)*T
      RP(K)=XYZ+.5
      SDF(K)=0
      EM(NEVENT)=TIME+DELT(K)
      RETURN
      END
```

Figure 5. Subroutine FORE1, Base Simulator

```
      SUBROUTINE RCPT(PIL,RCQ,QR,EM,TR,QH)
C     PURPOSE - TO RECEIVE INCOMING SHIPMENTS AND TO FILL BACKORDERS.
      DIMENSION PIL(3),RCQ(3),QR(3),EM(12),TR(3),QH(2),SS(2),TAU(3)
      COMMON DU(27),TIME,NEVENT,DUM(10),TAU
      INTEGER EM,TAU,TR,TIME,PIL,RCQ,QR,QH,SS,TOT
      J=NEVENT
C     K IDENTIFIES THE FACILITY.
      K=NEVENT-5
      PIL(K)=PIL(K)+RCQ(K)
      IF(K.EQ.3) GO TO 51
C     THE FOLLOWING STATEMENTS CHECK TO SEE IF ANOTHER SHIPMENT IS
C     ENROUTE, WHEN K IS A BRANCH FACILITY. IF SO, IT IS PLACED
C     IN RCQ AND ITS ARRIVAL TIME IS PLACED IN EM.
      IF(QR(K).EQ.0) GO TO 50
      RCQ(K)=QR(K)
      EM(J)=TR(K)
      QR(K)=0
      TR(K)=3000
      GO TO 59
C     END OF PREVIOUS COMMENT.
C     THE NUMBER 3000 AS USED IN THIS PROGRAM REPRESENTS ANY LARGE
C     NUMBER.
   50 RCQ(K)=0
      EM(J)=3000
      GO TO 59
   51 IF(PIL(K).GE.0) GO TO 55
C     THE FOLLOWING STATEMENTS DIVIDE A RECEIPT AT THE CENTRAL
C     WAREHOUSE WHICH IS INSUFFICIENT TO FILL ALL BACKORDERS.
C     RECEIPTS AT THE BRANCHES ARE SCHEDULED.
C     ZPY,TOT, AND INT ARE VARIABLES USED ONLY IN THE COMPUTATION
C     OF PROPORTIONATE SHARES.
C     SS(1) AND SS(2) ARE THE PROPORTIONATE SHARES.
      TOT=QH(1)+QH(2)
      ZPY=RCQ(3)*QH(1)
      REG=ZPY/TOT
      INT=REG+.5
      SS(1)=INT
      SS(2)=RCQ(3)-SS(1)
      DO 54 I=1,2
      L=I+5
      IF(SS(I).EQ.0) GO TO 54
      IF(RCQ(I).EQ.0) GO TO 52
      QR(I)=SS(I)
      TR(I)=TAU(I)+TIME
      GO TO 53
   52 RCQ(I)=SS(I)
      EM(L)=TAU(I)+TIME
```

(Continued)

```
   53 QH(I)=QH(I)-SS(I)
   54 CONTINUE
      GO TO 90
C     END OF PREVIOUS COMMENTS.
C     THE FOLLOWING STATEMENTS FILL BACKORDERS FOR THE BRANCHES
C     WHEN A RECEIPT AT THE CENTRAL WAREHOUSE IS SUFFICIENT TO
C     DO SO.
   55 DO 58 I=1,2
      L=I+5
      IF(QH(I).EQ.0) GO TO 58
      IF(RCQ(I).EQ.0) GO TO 56
      QR(I)=QH(I)
      TR(I)=TAU(I)+TIME
      GO TO 57
   56 RCQ(I)=QH(I)
      EM(L)=TAU(I)+TIME
   57 QH(I)=0
   58 CONTINUE
   90 IF(QR(3).EQ.0) GO TO 91
      RCQ(3)=QR(3)
      EM(8)=TR(3)
      QR(3)=0
      TR(3)=3000
      GO TO 59
   91 RCQ(3)=0
      EM(8)=3000
C     END OF PREVIOUS COMMENT.
   59 RETURN
      END
```

Figure 6. Subroutine RCPT, Base Simulator

A check is then made of QR(K) to see if another shipment is enroute (the second part of a split order). If QR(K) is not zero, its value is placed in RCQ(K), and its time of arrival is transferred from vector TR to the appropriate position in EM.

<u>Generating and Processing First-Echelon Demands</u>. The means and standard deviations of the input demand processes are read into the vectors MU and SD in the main program (Figure 10). The values in these vectors are brought into Subroutine DR01 (Figure 7), where first-echelon demands are processed.

```
      SUBROUTINE DR01(SDF,PIL,PIP,ORD,EM,NO)
C     PURPOSE - TO GENERATE DEMANDS, REVIEWS, AND ORDERS FOR
C     BRANCHES 1 AND 2.
      DIMENSION PIL(3),SDF(3),NO(3),ORD(3),RP(3),PIP(3),MU(2),SD(2),SM(3
     1),A(3),EM(12),D(3)
      COMMON DU(15),RP,SM,DUM(6),TIME,NEVENT,DUMMY(17),MU,A,D,C,SD
      COMMON/COM1/AVE,DEV
      REAL MU
      INTEGER TIME,EM,SDF,PIL,PIP,RP,ORD,Q
C     K IDENTIFIES THE FACILITY.
      K=NEVENT-8
      AVE=MU(K)
      DEV=SD(K)
      RV=0
      CALL XNORM(RV)
C     THIS SUBROUTINE GENERATES A REALIZATION FROM A RANDOM VARIABLE
C     DISTRIBUTED N(MU(K),SD(K)).
      JPQ=RV+.5
C     JPQ IS THE DEMAND.
      SDF(K)=SDF(K)+JPQ
      PIL(K)=PIL(K)-JPQ
      PIP(K)=PIP(K)-JPQ
      IF(PIP(K).GT.RP(K)) GO TO 60
      Q1=SQRT((2*SM(K)*A(K))/(D(K)*C))
      Q=Q1+.5
      ORD(K)=Q+(RP(K)-PIP(K))
      PIP(K)=PIP(K)+ORD(K)
C     THE NEXT TWO STATEMENTS SCHEDULE THE ARRIVAL OF THIS ORDER
C     AT THE CENTRAL WAREHOUSE.
      JJ=NEVENT+2
      EM(JJ)=TIME+1
      NO(K)=NO(K)+1
   60 EM(NEVENT)=TIME+1
      RETURN
      END
```

Figure 7. Subroutine DR01, Base Simulator

Within this subroutine, Subroutine XNORM (Figure 8) is called to generate the proper normal random variable (variable RV). This random variable is rounded to an integer value and placed in variable JPQ. JPQ is then added to the value in SDF(K), where K is the facility experiencing the demand, for later use in forecasting.

```
      SUBROUTINE XNORM(RV)
C     PURPOSE - TO GENERATE THE NORMAL RANDOM VARIABLE
C     WHICH IS ROUNDED TO THE INTEGER DEMAND IN DRO1.
      DIMENSION Y(12)
      COMMON DUMMY(45),IX
C     IX IS THE RANDOM NUMBER SEED.
      COMMON/COM1/AVE,DEV
      DO 110 I=1,12
      IY=IX*131075
      IF(IY) 100,101,101
  100 IY=IY+34359738367+1
  101 YFL=IY
      IX=IY
  110 Y(I)=YFL*(2.0)**(-35)
      PROD=0
      DO 111 I=1,12
  111 PROD=PROD+Y(I)
      Z=PROD-6
      RV=AVE+DEV*Z
      IF(RV.LT.0) GO TO 112
      GO TO 113
  112 RV=0
  113 RETURN
      END
```

Figure 8. Subroutine XNORM, Base Simulator

The net inventory, PIL(K), and the inventory position in PIP(K) are decremented by the value in JPQ. The value in PIP(K) is then compared to the reorder point in RP(K). If an order is appropriate, the EOQ formula is solved and the value, rounded to an integer, is placed in variable Q. The proper order quantity is computed and placed in ORD(K). The arrival of this order at the central warehouse is then scheduled for the following day by placing the value TIME+1 in the appropriate position in vector EM.

The process generator (Subroutine XNORM) produces the normal random variables by first producing 12 random numbers. These random numbers are then used to generate a single random variable. Negative random variables are possible with this generator and, if they occur, the negative value is replaced by zero.

<u>Order Processing at the Central Warehouse.</u> Subroutine DRO2 (Figure 9) is responsible for several actions.

```
      SUBROUTINE DRO2(PIL,PIP,EM,RCQ,QR,TR,QH,ORD,NO,SDF,UD)
C     PURPOSE - TO ACCEPT DEMANDS FROM BRANCHES AT THE CENTRAL WAREHOUSE
C     SHIP ORDERS IF STOCK IS ON HAND, REVIEW CENTRAL WAREHOUSE
C     INVENTORY POSITION, ORDER IF NECESSARY, AND SCHEDULE RECEIPTS.
      DIMENSION PIL(3),ORD(3),PIP(3),EM(12),RCQ(3),QR(3),TR(3),TAU(3),QH
     1(2),RP(3),SM(3),A(3),NO(3),SDF(3),D(3)
      COMMON DU(15),RP,SM,DUM(6),TIME,NEVENT,DUMM(10),TAU,DUMMY(6),A,D,C
      INTEGER EM,TAU,TIME,TR
      INTEGER PIL,PIP,RCQ,QR,QH,ORD,RP,UD,TOT,Q,SDF
      K=NEVENT
      L=K-2
      M=K-10
      N=K-5
      SDF(3)=SDF(3)+ORD(M)
      IF(K.EQ.11) GO TO 71
C     IF THIS SECTION IS ENTERED, THE ORDER IS FROM FACILITY 2.
   70 INT=PIL(3)
      PIL(3)=PIL(3)-ORD(M)
      PIP(3)=PIP(3)-ORD(M)
      GO TO 72
C     IF THE FOLLOWING SECTION IS ENTERED, THE ORDER IS FROM FACILITY 1.
   71 IF(EM(12).NE.TIME) GO TO 70
      IF(PIL(3).LE.0) GO TO 70
      TOT=ORD(1)+ORD(2)
      IF(PIL(3).GT.TOT) GO TO 70
C     IF THIS SECTION IS ENTERED, AN ORDER WILL ALSO ARRIVE ON THIS
C     DAY FROM FACILITY 2, AND INSUFFICIENT STOCK IS ON HAND TO FILL
C     BOTH ORDERS.  ZPY,REG,AND TOT ARE USED TO COMPUTE THAT PART OF
C     STOCK THAT WILL GO TO FACILITY 1.  INT IS THIS QUANTITY.
      ZPY=PIL(3)*ORD(1)
      REG=ZPY/TOT
      INT=REG+.5
      PIL(3)=PIL(3)-INT
      PIP(3)=PIP(3)-ORD(1)
C     UD IS PART OF ORDER FROM FACILITY 1 NOT SHIPPED.
C     INT AS USED HERE IS STARTING NET INVENTORY.
      UD=ORD(1)-INT
      GO TO 76
   72 IF(INT.LE.0) GO TO 74
```

(Continued)

```
C      IF THE STARTING NET INVENTORY WAS NEGATIVE OR ZERO,
C      THE TOTAL ORDER IS BACKLOGGED.
       IF(PIL(3).LT.0) GO TO 75
       IF(RCQ(M).EQ.0) GO TO 73
       QR(M)=ORD(M)
       TR(M)=TIME+TAU(M)
       GO TO 79
    73 RCQ(M)=ORD(M)
       EM(N)=TIME+TAU(M)
       GO TO 79
    74 QH(M)=QH(M)+ORD(M)
       GO TO 79
    75 IF(UD.EQ.0) GO TO 76
C      IF UD IS NOT ZERO, THEN THE ORDER IS FROM FACILITY 2.
C      AND AN ORDER HAS ALREADY ARRIVED THIS DAY FROM FACILITY 1.
       PIL(3)=PIL(3)-UD
       UD=0
    76 IF(RCQ(M).EQ.0) GO TO 77
       QR(M)=INT
       TR(M)=TIME+TAU(M)
       GO TO 78
    77 RCQ(M)=INT
       EM(N)=TIME+TAU(M)
    78 QH(M)=QH(M)+ORD(M)-INT
    79 EM(K)=3000
C      THE FOLLOWING SECTION REVIEWS INVENTORY AND ORDERS.
       IF(PIP(3).GT.RP(3)) GO TO 80
       Q1=SQRT((2*SM(3)*A(3))/(D(3)*C))
       Q=Q1+.5
       ORD(3)=Q+(RP(3)-PIP(3)
       NO(3)=NO(3)+1
       PIP(3)=PIP(3)+ORD(3)
       IF(RCQ(3).EQ.0) GO TO 81
       QR(3)=ORD(3)
       TR(3)=TIME+TAU(3)
       GO TO 80
    81 RCQ(3)=ORD(3)
       EM(8)=TIME+TAU(3)
    80 RETURN
       END
```

Figure 9. Subroutine DR02, Base Simulator

Its first function is to accept an order from a branch facility and add the value of this order to SDF(3), for later use in forecasting. PIL(3) and PIP(3) are then decremented by the order quantity ORD(M), where M identifies the facility that has ordered. If sufficient stock is on hand to completely fill the order, ORD(M) is placed in RCQ(M), and the scheduled arrival time, TIME+TAU(M), where TAU is a vector of lead times, is placed in vector EM. If an order cannot be completely filled, the value of the unfilled portion is stored in QH(M) until the central warehouse receives a shipment. At this time, the value of QH(M) will be placed in RCQ(M), if it is empty, and in QR(M), if it is not. If QR(M) is used, the scheduled arrival time will be placed in TR(M).

Before filling an order, this subroutine determines whether the other branch facility also has an order request scheduled for arrival on the same day. If such is the case, a further check is made to see if both orders can be satisfied from available stock. If not, each branch will be shipped a proportionate share of what is available.

After processing an order, the subroutine compares PIP(3) to RP(3) and orders, if appropriate. The ordering procedure is identical to the one used in Subroutine DR01. If an order is placed for the central warehouse, its arrival is scheduled for TIME+TAU(3). The order quantity is immediately placed in RCQ(3).

Main Program. The heart of the main program (Figure 10) is contained in those statements which drive the simulation through time. These operations were depicted in the macro flow diagram (Figure 2). The two key statements related to event selection and routing to the proper event subroutine were given earlier. After each event is executed, the stagement "GO TO 216" is used to loop the simulation back, so that Subroutine NEXT can again be called. This loop is exited only when the next event becomes Event 2 (NEVENT=2). When this occurs, the computed GO TO statement routes the simulation out of the loop to a series of statements which summarize statistics, print statistics, and then end the simulation.

```
C      ON THE FOLLOWING COMMENT CARDS I=1,...,3; J=1,...,2; K=1,...,12;
C      AIQ(I) = CUMULATIVE UNIT DAYS OF INVENTORY FOR FACILITY I
C      BACK(I) = CUMULATIVE UNIT DAYS OF BACKORDERS FOR FACILITY I
C      PIL(I) = CURRENT VALUE OF NET INVENTORY FOR FACILITY I
C      PIP(I) = CURRENT VALUE OF INVENTORY POSITION FOR FACILITY I
C      RP(I) = CURRENT VALUE OF THE REORDER POINT FOR FACILITY I
C      ORD(I) = VALUE OF THE LATEST QUANTITY ORDERED BY FACILITY I
C      RCQ(I) = VALUE OF THE QUANTITY IN THE NEXT EARLIEST SCHEDULED
C               RECEIPT AT FACILITY I
C      QR(I) = VALUE OF A QUANTITY SCHEDULED FOR RECEIPT AT FACILITY I
C              LATER THAN THE QUANTITY IN RCQ - THIS IS A TEMPORARY
C              HOLDING LOCATION AWAITING RCQ(I) TO BECOME EMPTY
C      TR(I) = SCHEDULED TIME OF ARRIVAL OF QUANTITY IN QR(I)
C      QH(J) = THAT PART OF ORDERS PLACED BY BRANCH FACILITY J THAT
C              CANNOT BE SATISFIED FROM STOCK ON HAND AT THE CENTRAL
C              WAREHOUSE (FACILITY 3)
C      SDF(I) = SUM OF DEMANDS AT FACILITY I OVER THE CURRENT FORECAST
C               INTERVAL
C      MU(J) = VALUE OF CONSTANT PART OF THE DEMAND PROCESS IMPOSED ON
C              BRANCH J
C      SD(J) = STANDARD DEVIATION OF NOISE IN THE DEMAND PROCESS IMPOSED
C              ON BRANCH J
C      ERT(I) = CURRENT SUM OF FORECAST ERRORS AT FACILITY I
C      SM(I) = LATEST FORECAST OF DEMAND OVER A FORECAST INTERVAL AT
C              FACILITY I
C      SMAD(I) = LATEST ESTIMATE OF THE MEAN ABSOLUTE DEVIATION OF
C                FORECAST ERROR AT FACILITY I
C      NEVENT = EVENT CURRENTLY SCHEDULED FOR EXECUTION
C      EM(K) = NEXT EARLIEST TIME OF EXECUTION FOR EACH EVENT
C      DELT(I) = FORECAST INTERVAL FOR FACILITY I
C      TAU(I) = REPLENISHMENT LEAD TIME FOR FACILITY I
C      TLAS = TIME OF LAST EVENT EXECUTED
C      TIME = TIME OF CURRENT EVENT BEING EXECUTED
C      TSTA = LENGTH OF SIMULATION WARM-UP PERIOD
C      TERM = TOTAL LENGTH OF SIMULATION RUN
C      CNTR(I) = CONTROL VALUE FOR COMPARISON WITH TRACKING SIGNAL AT
C                FACILITY I
C      ALO(I) = LOW LEVEL OF THE SMOOTHING CONSTANT FOR FACILITY I
C      AHI(I) = HIGH LEVEL OF THE SMOOTHING CONSTANT FOR FACILITY I
```

(Continued)

```
C     D(I) = INVENTORY CARRYING CHARGE RATE FOR FACILITY I
C     A(I) = FIXED ORDERING CHARGE FOR FACILITY I
C     C = UNIT VARIABLE COST OF AN ITEM
C     XXX = SAFETY FACTOR USED IN COMPUTING VALUE OF SAFETY STOCK
C     Q1 = COMPUTED VALUE OF THE ECONOMIC ORDER QUANTITY
C     NO(I) = A COUNTER FOR THE NUMBER OF ORDERS PLACED BY FACILITY I
C     IND(I) = A VARIABLE TO INDICATE WHETHER OR NOT THE PREVIOUS
C              FORECAST AT FACILITY I WAS IN CONTROL
C     ARRAY1 = STORAGE ARRAY FOR DAILY VALUES OF NET INVENTORY
C     ARRAY2 = STORAGE ARRAY FOR DAILY VALUES OF INVENTORY POSITION
C     ARRAY3 = STORAGE ARRAY FOR COMPUTED VALUES OF THE REORDER POINT
C     ARRAY4 = STORAGE ARRAY FOR COMPUTED VALUES OF THE FORECAST
C     ARRAY5 = STORAGE ARRAY FOR COMPUTED VALUES OF THE ESTIMATED
C              MEAN ABSOLUTE DEVIATION
C     ARRAY6 = STORAGE ARRAY FOR ORDER QUANTITIES
C     ARRAY7 = STORAGE ARRAY FOR TIMES AT WHICH ORDERS PLACED
C     KK(I), UD, AND KZY ARE VARIABLES USED AS COUNTERS FOR STORING
C     VALUES IN THE ARRAYS.
      DIMENSION MU(2),SD(2),TAU(3),ALO(3),AHI(3),DELT(3),A(3),AIQ(3),BAC
     1K(3),EM(12),ERT(3),IND(3),NO(3),QH(2),CNTR(3),KK(3),PIL(3),PIP(3),
     2SDF(3),SM(3),SMAD(3),RP(3),RCQ(3),QR(3),TR(3),ORD(3),ARRAY1(1171,3
     3),ARRAY2(1171,3),ARRAY3(168,3),ARRAY4(168,3),ARRAY5(168,3),ARRAY6(
     465,3),ARRAY7(65,3),D(3)
      REAL MU
      INTEGER TLAS,TIME,TERM,TSTA,EM,TAU,DELT,TR,UD,ARRAY6,ARRAY7
      INTEGER AIQ,BACK,QH,PIL,PIP,SDF,RP,RCQ,QR,ORD,ARRAY1,ARRAY2,ARRAY3
      COMMON PIL,EM,RP,SM,AIQ,BACK,TIME,NEVENT,TLAS,ALO,AHI,CNTR,TAU,DEL
     1T,IX,MU,A,D,C,SD,UD,XXX
      DO 200 I=1,3
  200 READ(5,201) TAU(I),DELT(I),ALO(I),AHI(I),A(I),CNTR(I),D(I)
  201 FORMAT(I3,I2,2F3.2,F6.2,F6.4,F5.4)
      READ(5,202) MU,SD
  202 FORMAT(4F5.2)
      READ(5,203) IX,TSTA,TERM,XXX,C
  203 FORMAT(I5,2I4,F5.2,F6.2)
      READ(5,204) SM,SMAD
  204 FORMAT(3F6.1,3F6.2)
      READ(5,205) RP
  205 FORMAT(3I4)
C     THE FOLLOWING OPERATIONS INITIALIZE VARIABLES
      DO 211 I=1,3
      Q1=SQRT((2*SM(I)*A(I))/(D(I)*C))
      PIL(I)=Q1+.5
  211 PIP(I)=PIL(I)
      DO 212 I=1,3
      AIQ(I)=0
      BACK(I)=0
      ERT(I)=0
      IND(I)=0
```

(Continued)

```
      NO(I)=0
      ORD(I)=0
      SDF(I)=0
      KK(I)=0
      QR(I)=0
      TR(I)=3000
C     THE VALUE 3000 WHENEVER USED IN THIS PROGRAM WILL SERVE AS A
C     LARGE NUMBER FOR CONTROL PURPOSES.
  212 RCQ(I)=0
      QH(1)=0
      QH(2)=0
      EM(1)=TSTA+1
      EM(2)=TERM
      DO 213 I=3,5
      K=I-2
  213 EM(I)=1+DELT(K)
      DO 214 I=6,8
  214 EM(I)=3000
      EM(9)=1
      EM(10)=1
      EM(11)=3000
      EM(12)=3000
      UD=0
      KZY=0
      NEVENT=1
      TIME=1
  216 TLAS=TIME
C     SUBROUTINE NEXT SELECTS THE NEXT EVENT FOR EXECUTION.
      CALL NEXT(NEVENT,TIME)
C     THE FOLLOWING STATEMENTS STORE VALUES OF NET INVENTORY AND
C     INVENTORY POSITION IN ARRAYS 1 AND 2.
      IF(TIME.EQ.TLAS) GO TO 218
      KZY=KZY+1
      DO 217 I=1,3
      ARRAY1(KZY,I)=PIL(I)
  217 ARRAY2(KZY,I)=PIP(I)
C     THE FOLLOWING STATEMENT ROUTES THE SIMULATION TO THE PROPER
C     SECTION FOR EXECUTION OF THE NEXT SCHEDULED EVENT
  218 GO TO (220,280,230,230,230,250,250,250,260,260,270,270),NEVENT
C     THIS SECTION CAUSES VALUES OF CUMULATIVE UNIT DAYS OF INVENTORY
C     AND BACKORDERS TO BE PRINTED AT END OF WARM-UP PERIOD.
  220 WRITE(6,221) AIQ,BACK,NO
  221 FORMAT(/,3(I8,4X))
      EM(1)=EM(1)+3000
      GO TO 216
C     THE FOLLOWING SECTION CALLS THE FORECASTING SUBROUTINE AND STORES
C     NEW VALUES OF RP,SM, AND SMAD IN ARRAYS.
  230 CALL FORE1(ERT,SM,SMAD,SDF,IND,RP,EM)
      I=NEVENT-2
```

(Continued)

```
      KK(I)=KK(I)+1
      J=KK(I)
      ARRAY3(J,I)=RP(I)
      ARRAY4(J,I)=SM(I)
      ARRAY5(J,I)=SMAD(I)
      GO TO 216
C     THE FOLLOWING SECTION EXECUTES A SCHEDULED RECEIPT OF STOCK.
  250 CALL RCPT(PIL,RCQ,QR,EM,TR,QH)
      GO TO 216
C     THE FOLLOWING SECTION CALLS THE SUBROUTINE TO GENERATE FIRST
C     ECHELON DEMANDS,REVIEWS, AND ORDERS, AND STORES ORDER QUANTITIES
C     AND TIMES IN ARRAYS 6 AND 7.
  260 IOU=NEVENT-8
      JZ=NO(IOU)
      CALL DRO1(SDF,PIL,PIP,ORD,EM,NO)
      IF(NO(IOU).EQ.JZ) GO TO 261
      I=NO(IOU)
      ARRAY6(I,IOU)=ORD(IOU)
      ARRAY7(I,IOU)=TIME
  261 GO TO 216
C     THE FOLLOWING SECTION CALLS THE SUBROUTINE TO FILL ORDERS FROM
C     THE BRANCHES, REVIEW INVENTORY POSITION, AND ORDER AT THE CENTRAL
C     WAREHOUSE.
  270 JZ=NO(3)
      CALL DRO2(PIL,PIP,EM,RCQ,QR,TR,QH,ORD,NO,SDF,UD)
      IF(NO(3).EQ.JZ) GO TO 271
      I=NO(3)
      ARRAY3(I,3)=ORD(3)
      ARRAY5(I,3)=TIME
  271 GO TO 216
C     THE FOLLOWING SECTION COMPUTES THE AVERAGE VALUES OF FORECASTS,
C     REORDER POINTS, ESTIMATES OF THE MEAN ABSOLUTE DEVIATION, AND
C     ORDER QUANTITIES, AND COMPUTES THE STANDARD DEVIATION OF ORDER
C     QUANTITIES FOR THE CENTRAL WAREHOUSE.
  280 SUM1=0
      SUM2=0
      SUM3=0
      DO 281 MN=3,38
      SUM1=SUM1+ARRAY3(MN,3)
      SUM2=SUM2+ARRAY4(MN,3)
  281 SUM3=SUM3+ARRAY5(MN,3)
      SUM1=SUM1/36
      SUM2=SUM2/36
      SUM3=SUM3/36
      LL=0
      DO 282 MN=1,65
      IF(ARRAY6(MN,3).EQ.0) GO TO 283
      LL=LL+1
  282 CONTINUE
```

(Continued)

```
  283 LIFT=1
      DO 284 N=1,LL
      IF(ARRAY7(N,3).GT.90) GO TO 285
      LIFT=LIFT+1
  284 CONTINUE
  285 SUM4=0
      SUM5=0
      DO 286 N=LIFT,LL
      SUM4=SUM4+ARRAY6(N,3)
  286 SUM5=SUM5+ARRAY6(N,3)**2
      EEK=SUM4**2/(LL-LIFT+1)
      SUM5=SQRT((SUM5-EEK)/(LL-LIFT))
      SUM4=SUM4/(LL-LIFT+1)
      WRITE(6,290) TLAS,TIME,IX
  290 FORMAT(/,3X,I5,3X,I5,3X,I15)
      WRITE(6,291) ((ARRAY1(J,K),ARRAY2(J,K),K=1,3),J=1,1170)
  291 FORMAT(/,6(3X,I7))
      WRITE(6,292) ((ARRAY3(J,K),ARRAY4(J,K),ARRAY5(J,K),K=1,3),J=1,167)
  292 FORMAT(/,3(1X,I5,1X,F7.2,1X,F6.2))
      WRITE(6,293) ((ARRAY6(J,K),ARRAY7(J,K),K=1,3),J=1,65)
  293 FORMAT(/,6(3X,I5))
      WRITE(6,294) SUM1,SUM2,SUM3,SUM4,SUM5
  294 FORMAT(/,2X,F8.2,2X,F8.2,2X,F7.2,2X,F8.2,2X,F7.2)
      END
```

Figure 10. Main Program, Base Simulator

Use of Procedure 2 in the Base Simulator. To use Procedure 2 for second-echelon forecasting, only a simple adjustment in the base simulator is necessary. Recall that SDF(3) contains the information on which second-echelon forecasts are based. To implement Procedure 2, the following statement is removed from Subroutine DRO2:

SDF(3) = SDF(3) + ORD(M)

Then, the following statement is placed in Subroutine DRO1:

SDF(3) = SDF(3) + JPQ

This statement would be placed immediately following the statement:

SDF(K) = SDF(K) + JPQ

The effect of these actions is to increment SDF(3) each time a demand is generated at a branch facility.

Use of Procedure 3 in the Base Simulator. To replace Procedure 1 with Procedure 3 requires several actions. A separate subroutine for second-echelon forecasts, Subroutine FORE2, will now be used (Figure 11).

```
      SUBROUTINE FORE2(SM,SMAD,RP,EM)
C     PURPOSE - TO UPDATE FORECASTS AND SET THE REORDER POINT FOR THE
C     CENTRAL WAREHOUSE.
      DIMENSION SDF(3),SM(3),SMAD(3),RP(3),TAU(3),
     1DELT(3),EM(12),VAR(2)
      COMMON DUMMY(27),TIME,DUM(11),TAU,DELT,DUMST(13),XXX
      INTEGER SDF,RP,EM,TAU,DELT,TIME
C     THE FOLLOWING STATEMENTS COMPUTE THE MEAN ABSOLUTE DEVIATION.
      V3=0
      DO 40 I=1,2
      VAR(I)=(1.25*SMAD(I))**2
   40 V3=V3+VAR(I)*DELT(3)/DELT(I)
      SMAD(3)=.8*SQRT(V3)
C     THE FOLLOWING STATEMENTS COMPUTE THE FORECAST AND REORDER POINT.
   41 SM(3)=SM(1)*DELT(3)/DELT(1)+SM(2)*DELT(3)/DELT(2)
      UAT=TAU(3)
      TLED=DELT(3)
      T=UAT/TLED
      XYZ=SM(3)*T+XXX*1.25*SMAD(3)*T
      RP(3)=XYZ+.5
      EM(5)=TIME+DELT(3)
      RETURN
      END
```

Figure 11. Subroutine FORE2, Base Simulator

The value of SDF(3) becomes meaningless, since second-echelon forecasts are achieved by combining the first-echelon forecasts. These forecasts are combined in the following statement:

```
SM(3) = SM(1)*DELT(3)/DELT(1) + SM(2)*DELT(3)/DELT(2)
```

DELT is a vector containing forecast intervals for the three facilities. The estimated values of the mean absolute deviation at each branch are changed to estimates of the variance of forecast errors. These variances are then extended over the forecast interval of the central warehouse and added to yield an estimate of the variance of forecast errors for the central warehouse. This estimated variance is then changed to an estimate of the mean absolute deviation.

The main program must be changed to route Event 5, a forecast for the central warehouse, to Subroutine FORE2. This is done by changing the computed GO TO statement as follows:

```
GO TO (220,280,230,230,240,250,250,250,260,260,
       270,270)NEVENT
```

A section must be added with label 240 to call Subroutine FORE2. Placement of this section is shown below.

```
230 CALL FORE1(ERT,SM,SMAD,SDF,IND,RP,EM)
                        .
                        .
                        .
    GO TO 216
240 CALL FORE2(SM,SMAD,RP,EM)
    KK(3)=KK(3)+1
    J=KK(3)
    ARRAY3(J,3)=RP(3)
    ARRAY4(J,3)=SM(3)
    ARRAY5(J,3)=SMAD(3)
    GO TO 216
250 CALL RCPT(PIL,RCQ,QR,EM,TR,QH)
                        .
                        .
                        .
```

Trend Simulator (Procedure 1)

This simulator uses a trend forecasting model and forecasts second-echelon demand based upon first-echelon orders. This program is constructed by adjusting Subroutines FORE1,DRO1, and DRO2, and the main program of the base simulator.

Forecasting. Figure 12 depicts the new version of Subroutine FORE1. No longer can the forecast of demand over a lead time be expressed as a multiple of the forecast over a forecast interval. In this program, two separate variables are used for these values. The variable SM in the base simulator is not used. The following new variables appear in this subroutine; each is a vector with three components, one for each facility.

SM1 = First smoothed statistic.

SM2 = Second smoothed statistic

CO1 = Estimate of the constant component in the forecast equation.

CO2 = Estimate of the trend component in the forecast equation.

```
      SUBROUTINE FORE1(ERT,SMAD,SDF,IND,RP,EM,SM1,SM2,CO1,CO2,FO,FOF)
C     PURPOSE - TO UPDATE FORECASTS.
C     THE VARIABLE FOC IS A TEMPORARY STORAGE LOCATION USED IN SUMMING.
      DIMENSION SDF(3),FO(3),ERT(3),SMAD(3),ALO(3),AHI(3),CNTR(3),IND(3)
     1,RP(3),TAU(3),DELT(3),EM(12),SM1(3),SM2(3),CO1(3),CO2(3),FOF(3)
      COMMON DUM(27),TIME,NEVENT,TLAS,ALO,AHI,CNTR,TAU,DELT,DU(13),XXX
      INTEGER EM,TAU,TIME,DELT,SDF,RP
      K=NEVENT-2
      SUM=SDF(K)
      ERR=SUM-FOF(K)
      ERT(K)=ERT(K)+ERR
      SMAD(K)=ALO(K)*ABS(ERR)+(1-ALO(K))*SMAD(K)
      TS=ABS(ERT(K))/SMAD(K)
      IF(TS.GT.CNTR(K)) GO TO 30
      IND(K)=0
      GO TO 31
   30 IND(K)=IND(K)+1
      IF(IND(K).LE.1) GO TO 31
      ERT(K)=0
      SM1(K)=AHI(K)*SUM+(1-AHI(K))*SM1(K)
      SM2(K)=AHI(K)*SM1(K)+(1-AHI(K))*SM2(K)
      CO1(K)=2*SM1(K)-SM2(K)
      CO2(K)=(SM1(K)-SM2(K))*AHI(K)/(1-AHI(K))
      GO TO 32
   31 SM1(K)=ALO(K)*SUM+(1-ALO(K))*SM1(K)
      SM2(K)=ALO(K)*SM1(K)+(1-ALO(K))*SM2(K)
      CO1(K)=2*SM1(K)-SM2(K)
      CO2(K)=(SM1(K)-SM2(K))*ALO(K)/(1-ALO(K))
   32 CONTINUE
      FOF(K)=CO1(K)+CO2(K)
      UAT=TAU(K)
      QUO=UAT/DELT(K)
      KUO=TAU(K)/DELT(K)
      FOC=0
      DO 33 I=1,KUO
   33 FOC=FOC+CO1(K)+CO2(K)*I
      FO(K)=FOC+FOF(K)*(QUO-KUO)
      XYZ=FO(K)+XXX*1.25*SMAD(K)*TAU(K)/DELT(K)
      RP(K)=XYZ+.5
      SDF(K)=0
      EM(NEVENT)=TIME+DELT(K)
      RETURN
      END
```

Figure 12. Subroutine FORE1, Trend Simulator

FOF = Forecast of the demand rate one forecast interval into the future.

FO = Forecast of the demand over a lead time.

There is no change in the initial part of the subroutine which computes the revised estimate of the mean absolute deviation and the tracking signal. In the FORTRAN statements which follow this part, FOF is equivalent to $\hat{x}_{t1}$ in Equation 2-7, and FO, KUO, and (QUO-KUO) correspond to $\hat{H}$, n, and s, respectively, in Equation 2-8.

Generating and Processing First-Echelon Demands. In Chapter III, the trend component was selected to reflect a growth in one year of one-half the value of the constant component. One parameter, GR, and one variable, ACC, are used in conjunction with the normal variate generator to generate this process. GR is the value of the desired daily growth. Variable ACC is the value of this accumulated growth to date, at any time. In the revised Subroutine DR01 (Figure 13), GR is added each day to ACC. Subroutine XNORM (unchanged) is then called, producing the normally distributed random variable, RV. Before being rounded to an integer value, RV is increased by ACC. In this way, a linear demand process with a normally distributed noise is achieved.

The expected demand over a lead time is used in the EOQ formula; thus, the carrying-charge rate, D, must now reflect a lead time, rather than a forecast interval, as a basic unit of time.

Second-Echelon Demand. Few changes are needed in Subroutine DR02. In the COMMON and DIMENSION statements, the vector FO replaces SM. In the EOQ formula, FO is used, and the comment in the last paragraph applies here, as well.

```
      SUBROUTINE DR01(SDF,PIL,PIP,ORD,EM,NO)
C     PURPOSE - TO GENERATE DEMANDS, REVIEWS, AND ORDERS FOR
C     BRANCHES 1 AND 2.
      DIMENSION PIL(3),SDF(3),NO(3),ORD(3),RP(3),PIP(3),MU(2),SD(2),FO(3
     1),A(3),EM(12),D(3),GR(2),ACC(2)
      COMMON DU(15),RP,FO,DUM(6),TIME,NEVENT,DUMMY(17),MU,A,D,C,SD
      COMMON/COM1/AVE,DEV
      COMMON/COM2/GR,ACC
      REAL MU
      INTEGER TIME,EM,SDF,PIL,PIP,RP,ORD,Q
      K=NEVENT-8
      ACC(K)=ACC(K)+GR(K)
      AVE=MU(K)
      DEV=SD(K)
      RV=0
      CALL XNORM(RV)
C     THIS SUBROUTINE GENERATES A REALIZATION FROM A RANDOM VARIABLE
C     DISTRIBUTED N(MU(K),SD(K)).
      RV=RV+ACC(K)
      JPQ=RV+.5
      SDF(K)=SDF(K)+JPQ
      PIL(K)=PIL(K)-JPQ
      PIP(K)=PIP(K)-JPQ
      IF(PIP(K).GT.RP(K)) GO TO 60
      Q1=SQRT((2*FO(K)*A(K))/(D(K)*C))
      Q=Q1+.5
      ORD(K)=Q+(RP(K)-PIP(K))
      PIP(K)=PIP(K)+ORD(K)
      JJ=NEVENT+2
      EM(JJ)=TIME+1
      NO(K)=NO(K)+1
   60 EM(NEVENT)=TIME+1
      RETURN
      END
```

Figure 13. Subroutine DR01, Trend Simulator

Main Program. Changes in the main program of the base simulator are of a control or administrative type. The statements which require revision are used to read input data, initialize variables, and store statistics in arrays.

To incorporate the new variables included in the revised subroutines, the following statements replace comparable statements in the base simulator.

```
      DIMENSION MU(2),SD(2),TAU(3),ALO(3),AHI(3),DELT(3),A(3),AIQ(3),BAC
     1K(3),EM(12),ERT(3),IND(3),NO(3),QH(2),CNTR(3),KK(3),PIL(3),PIP(3),
     2SDF(3),FO(3),SMAD(3),RP(3),RCQ(3),QR(3),TR(3),ORD(3),ARRAY1(1171,3
     3),ARRAY2(1171,3),ARRAY3(168,3),ARRAY4(168,3),ARRAY5(168,3),ARRAY6(
     465,3),ARRAY7(65,3),D(3),GR(2),ACC(2),SM1(3),SM2(3),CO1(3),CO2(3),F
     5OF(3)
                                         .
                                         .
                                         .
      COMMON PIL,EM,RP,FO,AIQ,BACK,TIME,NEVENT,TLAS,ALO,AHI,CNTR,TAU,DEL
     1T,IX,MU,A,D,C,SD,UD,XXX
                                         .
                                         .
                                         .
      READ(5,205) FO,SMAD
                                         .
                                         .
                                         .
      Q1=SQRT((2*FO(I)*A(I))/(D(I)*C))
                                         .
                                         .
                                         .
  230 CALL FORE1(ERT,SMAD,SDF,IND,RP,EM,SM1,SM2,CO1,CO2,FO,FOF)
                                         .
                                         .
                                         .
      ARRAY4(J,I)=FOF(I)
```

The following statements are added at appropriate locations in the main program to read-in values and initialize variables.

```
      COMMON/COM2/GR,ACC
                  .
                  .
                  .
      READ(5,206) GR,CO1,CO2
  206 FORMAT(2F6.5,3F6.1,3F6.2)
      READ(5,207) SM1,SM2
  207 FORMAT(6F7.2)
                  .
                  .
                  .
      FOF(I)=CO1(I)
                  .
                  .
                  .
      ACC(1)=0
      ACC(2)=0
```

FOF is initialized as CO1 since the trend effect begins with the start of the simulation.

Use of Procedure 2 in the Trend Simulator. This adjustment is identical to the change made in the base simulator to use Procedure 2 for second-echelon forecasting. (See Page 67.)

Use of Procedure 3 in the Trend Simulator. To replace Procedure 1 with Procedure 3 requires actions similar to those used on the base simulator to implement this change. In the main program, the same revised GO TO statement is used. (See Page 69.) A section labelled 240 is added as follows:

```
230 CALL FORE1(ERT,SMAD,SDF,IND,RP,EM,SM1,SM2,CO1,CO2,FO,FOF)
                             .
                             .
                             .
    GO TO 216
240 CALL FORE2(FO,SMAD,FOF,RP,EM,CO1,CO2)
    KK(3)=KK(3)+1
    J=KK(3)
    ARRAY3(J,3)=RP(3)
    ARRAY4(J,3)=FOF(3)
    ARRAY5(J,3)=SMAD(3)
    GO TO 216
250 CALL RCPT(PIL,RCQ,QR,EM,TR,QH)
                             .
                             .
                             .
```

Subroutine FORE2 is revised as shown in Figure 14. With the trend model, second-echelon forecasts can no longer be determined by simply combining the first-echelon forecasts. Instead, the estimates of the coefficients for the branch forecast equations (CO1 and CO2) are used. The central warehouse, using these branch coefficients, computes forecasts over its own forecast interval and lead time for each branch facility. The values so obtained are then added to give the forecasts for the central warehouse.

```
      SUBROUTINE FORE2(FO,SMAD,FOF,RP,EM,CO1,CO2)
C     PURPOSE - TO UPDATE FORECASTS AND SET THE REORDER POINT FOR THE
C     CENTRAL WAREHOUSE.
      DIMENSION FO(3),SMAD(3),RP(3),TAU(3),DELT(3),EM(12),VAR(2),CO1(3),
     1CO2(3),FOF(3)
      COMMON DUMMY(27),TIME,DUM(11),TAU,DELT,DUMST(13),XXX
      INTEGER RP,EM,TAU,DELT,TIME
C     THE FOLLOWING STATEMENTS COMBINE FIRST-ECHELON ESTIMATES TO
C     OBTAIN THE CENTRAL WAREHOUSE ESTIMATE OF SMAD.
      V3=0
      DO 40 I=1,2
      VAR(I)=(1.25*SMAD(I))**2
   40 V3=V3+VAR(I)*DELT(3)/DELT(I)
      SMAD(3)=.8*SQRT(V3)
C     THE FOLLOWING STATEMENTS COMPUTE THE FORECAST OVER THE NEXT
C     FORECAST INTERVAL.
      FOF(3)=0
      TLED=DELT(3)
      DO 42 J=1,2
      QUO=TLED/DELT(J)
      KUO=DELT(3)/DELT(J)
      FOC=0
      DO 41 I=1,KUO
   41 FOC=FOC+CO1(J)+CO2(J)*I
   42 FOF(3)=FOF(3)+FOC+(CO1(J)+CO2(J))*(QUO-KUO)
C     THE FOLLOWING STATEMENTS COMPUTE THE FORECAST OVER A LEAD TIME.
      FO(3)=0
      UAT=TAU(3)
      DO 44 J=1,2
      QUO=UAT/DELT(J)
      KUO=TAU(3)/DELT(J)
      FOC=0
      DO 43 I=1,KUO
   43 FOC=FOC+CO1(J)+CO2(J)*I
   44 FO(3)=FO(3)+FOC+(CO1(J)+CO2(J))*(QUO-KUO)
C     THE FOLLOWING TWO STATEMENTS COMPUTE THE REORDER POINT.
      XYZ=FO(3)+XXX*1.25*SMAD(3)*TAU(3)/DELT(3)
      RP(3)=XYZ+.5
      EM(5)=TIME+DELT(3)
      RETURN
      END
```

Figure 14. Subroutine FORE2, Trend Simulator

Seasonal Simulator (Procedure 1)

The seasonal forecast model used in this simulator has a linear trend effect and a multiplicative seasonal effect. The mathematical expressions for this model, presented in Chapter II, are straightforward; however, simulation of the model is complicated by the necessity to deal with two different time references. The first of these is normal simulation time (contained in the variable TIME in the base simulator); the second is seasonal cycle time. Whenever an event occurs which must reference points in the seasonal cycle, simulation time must be translated into cycle time. In addition, appropriate seasonal factors must be selected from tables to correspond with these points in the cycle. Most of the changes made in constructing the seasonal simulator from the base simulator deal with these complications.

The problem of setting the proper cycle time is addressed in the main program. The problem of selecting proper seasonal factors is handled in the subroutines. As with the previous simulation programs, changes in the subroutines are discussed prior to changes in the main program.

The seasonal factors are initialized to correspond to the factors used in generating the demands, and they are not updated in the simulations.

Forecasting at the First Echelon. The first-echelon facilities make forecasts every seven days. For ease in constructing the simulator, the six-month seasonal cycle will be viewed at this level as 182 days in length. The seasonal pattern for each facility is traced with

13 seasonal factors which are stored in the matrix F(13,2); thus, each factor applies to a 14-day period.

The variable KLOK1 contains the time with reference to a cycle length, and its value is assigned in the main program. This value may not exceed 183 (one greater than the cycle length, since forecasts are made on data from the previous 7 days).

Figure 15 shows Subroutine FORE1, revised to use the seasonal model. The following variables are the key values in the forecasting equations. Each, except F, which was explained above, is a vector with three components, but only the first two components (for the first-echelon facilities) are actually used in this subroutine.

SM = Smoothed statistic (constant factor).

R = Trend factor.

F = Seasonal factors.

ALO = Lower level of the smoothing constant used to update SM.

AHI = Higher level of the smoothing constant used to update SM.

WLO = Lower level of the smoothing constant used to update R.

WHI = Higher level of the smoothing constant used to update R.

FOF = Forecast of the demand rate one forecast interval into the future.

FO = Forecast of demand over the next lead time.

The value in KLOK1 identifies the seasonal factor to be used in the expression which updates SM. The seasonal factors to be used in computing FOF and FO are identified by adding DELT and TAU individually to KLOK1. The values placed in integer variables KK, L, and N locate these factors in the table (matrix F).

```
      SUBROUTINE FORE1(ERT,SMAD,SDF,IND,RP,EM,SM,F,R,FO,FOF)
C     PURPOSE - TO UPDATE FORECASTS FOR FIRST-ECHELON FACILITIES.
      DIMENSION SDF(3),FO(3),ERT(3),SMAD(3),ALO(3),AHI(3),CNTR(3),IND(3)
     1,RP(3),TAU(3),DELT(3),EM(12),SM(3),R(3),F(13,2),WLO(3),WHI(3),FOF(
     23)
      COMMON DUMM(27),TIME,NEVENT,TLAS,ALO,AHI,CNTR,TAU,DELT,DU(13),XXX
      COMMON/COM3/WHI,WLO,KLOK1,KLOK2
      INTEGER EM,TAU,TIME,DELT,SDF,RP
      K=NEVENT-2
      SUM=SDF(K)
      ERR=SUM-FOF(K)
      ERT(K)=ERT(K)+ERR
      SMAD(K)=ALO(K)*ABS(ERR)+(1-ALO(K))*SMAD(K)
      TS=ABS(ERT(K))/SMAD(K)
C     LAMP IDENTIFIES THE POINT IN TIME ON THE SEASONAL CYCLE ONE
C     FORECAST INTERVAL INTO THE FUTURE.
C     NAP IDENTIFIES THE POINT IN TIME ON THE SEASONAL CYCLE ONE
C     LEAD TIME INTO THE FUTURE.
C     KK IS THE SEQUENCE NUMBER OF THE FACTOR IN TABLE F USED TO
C     UPDATE SM.
C     L IS THE SEQUENCE NUMBER OF THE FACTOR IN TABLE F USED TO
C     COMPUTE FOF.
C     N IS THE SEQUENCE NUMBER OF THE FACTOR IN TABLE F USED TO
C     COMPUTE FO.
C     IGG IS A VARIABLE USED FOR COMPARISON WITH LAMP AND NAP TO
C     ASSIGN VALUES TO L AND N.
      LAMP=KLOK1+DELT(K)
      NAP=KLOK1+TAU(K)
      KK=1
      DO 30 I=15,183,14
      IF(KLOK1.LE.I) GO TO 31
   30 KK=KK+1
   31 L=KK
      N=KK
      IGG=14*KK+1
      IF(LAMP.GT.IGG) L=L+1
      IF(NAP.GT.IGG) N=N+1
      IF(L.GT.13) L=L-13
      IF(N.GT.13) N=N-13
      IF(TS.LE.CNTR(K)) GO TO 32
      IND(K)=IND(K)+1
      IF(IND(K).LE.1) GO TO 33
      ERT(K)=0
      SP=SM(K)
      SM(K)=AHI(K)*SDF(K)/F(KK,K)+(1-AHI(K))*(SP+R(K))
      R(K)=WHI(K)*(SM(K)-SP)+(1-WHI(K))*R(K)
      GO TO 34
   32 IND(K)=0
   33 SP=SM(K)
```

(Continued)

```
      SM(K)=ALO(K)*SDF(K)/F(KK,K)+(1-ALO(K))*(SP+R(K))
      R(K)=WLO(K)*(SM(K)-SP)+(1-WLO(K))*R(K)
   34 FOF(K)=(SM(K)+R(K))*F(L,K)
      FO(K)=FOF(K)+(SM(K)+2*R(K))*F(N,K)
      XYZ=FO(K)+XXX*1.25*SMAD(K)*TAU(K)/DELT(K)
      RP(K)=XYZ+.5
      SDF(K)=0
      EM(NEVENT)=TIME+DELT(K)
      RETURN
      END
```

Figure 15. Subroutine FORE1, Seasonal Simulator

Forecasting at the Second Echelon. A separate subroutine is used in this simulator to make forecasts for the central warehouse. The central warehouse forecasts demand every 30 days; therefore, for ease in simulation, the six-month cycle will be viewed as 180 days in length. Here, the seasonal pattern is traced with six factors which are stored in the vector FF; thus, each factor applies to a 30-day period.

The operation of this subroutine (Figure 16) is identical to the operation of Subroutine FORE1 in Figure 15, except that the table of seasonal factors is of different size. KLOK2 is used in this subroutine to reference current time with respect to a point in the seasonal cycle. The value of KLOK2 may not exceed 181.

```
      SUBROUTINE FORE2(ERT,SMAD,SDF,IND,RP,EM,SM,FF,R,FO,FOF)
C     PURPOSE - TO UPDATE FORECASTS AT THE CENTRAL WAREHOUSE.
C     COMMENT CARDS OF FIGURE 15 APPLY.
      DIMENSION SDF(3),FO(3),ERT(3),SMAD(3),ALO(3),AHI(3),CNTR(3),IND(3)
     1,RP(3),TAU(3),DELT(3),EM(12),SM(3),R(3),FF(6),WHI(3),WLO(3),FOF(3)
      COMMON DUMM(27),TIME,NEVENT,TLAS,ALO,AHI,CNTR,TAU,DELT,DU(13),XXX
      COMMON/COM3/WHI,WLO,KLOK1,KLOK2
      INTEGER EM,TAU,TIME,DELT,SDF,RP
      SUM=SDF(3)
```

(Continued)

```
      ERR=SUM-FOF(3)
      ERT(3)=ERT(3)+ERR
      SMAD(3)=ALO(3)*ABS(ERR)+(1-ALO(3))*SMAD(3)
      TS=ABS(ERT(3))/SMAD(3)
      LAMP=KLOK2+DELT(3)
      NAP=KLOK2+TAU(3)
      KK=1
      DO 30 I=31,181,30
      IF(KLOK2.LE.I) GO TO 31
   30 KK=KK+1
   31 L=KK
      N=KK
      IGG=30*KK+1
      IF(LAMP.GT.IGG) L=L+1
      IF(NAP.GT.IGG) N=N+1
      IF(L.GT.6) L=L-6
      IF(N.GT.6) N=N-6
      IF(TS.LE.CNTR(3)) GO TO 32
      IND(3)=IND(3)+1
      IF(IND(3).LE.1) GO TO 33
      ERT(3)=0
      SP=SM(3)
      SM(3)=AHI(3)*SDF(3)/FF(KK)+(1-AHI(3))*(SP+R(3))
      R(3)=WHI(3)*(SM(3)-SP)+(1-WHI(3))*R(3)
      GO TO 34
   32 IND(3)=0
   33 SP=SM(3)
      SM(3)=ALO(3)*SDF(3)/FF(KK)+(1-ALO(3))*(SP+R(3))
      R(3)=WLO(3)*(SM(3)-SP)+(1-WLO(3))*R(3)
   34 FOF(3)=(SM(3)+R(3))*FF(L)
      FO(3)=FOF(3)
      XYZ=FO(3)+XXX*1.25*SMAD(3)*TAU(3)/DELT(3)
      RP(3)=XYZ+.5
      SDF(3)=0
      EM(NEVENT)=TIME+DELT(3)
      RETURN
      END
```

Figure 16. Subroutine FORE2, Seasonal Simulator

<u>Generating and Processing First-Echelon Demands</u>. Figure 17 shows Subroutine DR01, revised to generate a seasonal pattern in demand. The parameter GR and the variable ACC are used in this model in the same manner as they were in the trend simulator.

```
      SUBROUTINE DR01(SDF,PIL,PIP,ORD,EM,NO)
C     PURPOSE - TO GENERATE DEMANDS, REVIEWS, AND ORDERS FOR
C     BRANCHES 1 AND 2.
      DIMENSION PIL(3),SDF(3),NO(3),ORD(3),RP(3),PIP(3),MU(2),SD(2),FO(3
     1),A(3),EM(12),D(3),GR(2),ACC(2),FFIL(20)
      COMMON DU(15),RP,FO,DUM(6),TIME,NEVENT,DUMMY(17),MU,A,D,C,SD
      COMMON/COM1/AVE,DEV
      COMMON/COM2/GR,ACC
      COMMON/COM4/KLOK3,FFIL
      REAL MU
      INTEGER TIME,EM,SDF,PIL,PIP,RP,ORD,Q
      K=NEVENT-8
      ACC(K)=ACC(K)+GR(K)
      AVE=MU(K)
      DEV=SD(K)
      RV=0
C     NR IS THE SEQUENCE NUMBER OF THE CORRECT FACTOR FROM FFIL TO USE.
      NR=1
      DO 61 I=9,180,9
      IF(KLOK3.LE.I) GO TO 62
   61 NR=NR+1
   62 CALL XNORM(RV)
C     THIS SUBROUTINE GENERATES A REALIZATION FROM A RANDOM VARIABLE
C     DISTRIBUTED N(MU(K),SD(K)).
      RV=(RV+ACC(K))*FFIL(NR)
      JPQ=RV+.5
      SDF(K)=SDF(K)+JPQ
      PIL(K)=PIL(K)-JPQ
      PIP(K)=PIP(K)-JPQ
      IF(PIP(K).GT.RP(K)) GO TO 60
      Q1=SQRT((2*FO(K)*A(K))/(D(K)*C))
      Q=Q1+.5
      ORD(K)=Q+(RP(K)-PIP(K))
      PIP(K)=PIP(K)+ORD(K)
      JJ=NEVENT+2
      EM(JJ)=TIME+1
      NO(K)=NO(K)+1
   60 EM(NEVENT)=TIME+1
      RETURN
      END
```

Figure 17. Subroutine DR01, Seasonal Simulator

A table of seasonal factors, FFIL, is provided. The factors are based upon a 180-day cycle of the sine curve, with an amplitude of one-half the value of the constant component in demand. A value is assigned to

KLOK3 in the main program, and this value is used in the subroutine to identify the correct component of FFIL to use.

After the normal random variable, RV, is generated in Subroutine XNORM, it is increased by ACC and, then, multiplied by the appropriate seasonal factor, before being rounded to an integer value.

It should be noted that FO is again used in the EOQ formula, both in this subroutine and in Subroutine DRO2.

Main Program. In Figure 18, the main program for the seasonal simulator is given without comment cards. Variables in this program, which do not appear in the base simulator and which have not been previously defined, are explained. Many of the additions or changes made in constructing this main program from the main program of the base simulator should require no explanation. Some of these statements declare variables and arrays, read-in data, and initialize variables. Others appeared in and were discussed with the trend simulator.

The proper cycle times are set in the labelled sections in which Subroutines FORE1, FORE2, and DRO1 are called. These sections are labelled 230, 240, and 260, respectively. The method is the same in each and will only be discussed for the first.

The cycle length (182 days) is placed in MARK1. The variable JES1 is initialized as zero, and the variable KONT1 is initialized as one. Variable MUTT is assigned a value which is compared to the simulation time, TIME. As long as TIME is less than or equal to MARK1+1, cycle time in KLOK1 is set equal to TIME. When simulation time becomes 184, KONT1 is incremented by one, causing KLOK1 to receive a value of

(184-182). This sequence continues throughout the simulation and restricts the value of KLOK1 to 183 or less.

```
      DIMENSION MU(2),SD(2),TAU(3),ALO(3),AHI(3),DELT(3),A(3),AIQ(3),BAC
     1K(3),EM(12),ERT(3),IND(3),NO(3),QH(2),CNTR(3),KK(3),PIL(3),PIP(3),
     2SDF(3),FO(3),SMAD(3),RP(3),RCQ(3),QR(3),TR(3),ORD(3),FOF(3),ARRAY1
     3(1171,3),ARRAY2(1171,3),ARRAY3(168,3),ARRAY4(168,3),ARRAY5(168,3),
     4ARRAY6(65,3),ARRAY7(65,3),D(3),GR(2),ACC(2),SM(3),R(3),FFIL(20),F(
     513,2),FF(6),WHI(3),WLO(3)
      REAL MU
      INTEGER TLAS,TIME,TERM,TSTA,EM,TAU,DELT,TR,UD,ARRAY6,ARRAY7
      INTEGER AIQ,BACK,QH,PIL,PIP,SDF,RP,RCQ,QR,ORD,ARRAY1,ARRAY2,ARRAY3
      COMMON PIL,EM,RP,FO,AIQ,BACK,TIME,NEVENT,TLAS,ALO,AHI,CNTR,TAU,DEL
     1T,IX,MU,A,D,C,SD,UD,XXX
      COMMON/COM2/GR,ACC
      COMMON/COM3/WHI,WLO,KLOK1,KLOK2
      COMMON/COM4/KLOK3,FFIL
      DO 200 I=1,3
  200 READ(5,201) TAU(I),DELT(I),ALO(I),AHI(I),A(I),CNTR(I),D(I)
  201 FORMAT(I3,I2,2F3.2,F6.2,F6.4,F5.4)
      READ(5,203) MU,SD,GR,WLO,WHI
  203 FORMAT(4F5.2,2F6.5,6F3.2)
      READ(5,204) IX,TSTA,TERM,XXX,C
  204 FORMAT(I5,2I4,F5.2,F6.2)
      READ(5,205) FO,SM,R,SMAD
  205 FORMAT(6F6.1,6F6.2)
      READ(5,206) RP
  206 FORMAT(3I4)
      READ(5,207) F
  207 FORMAT(26F5.3)
      READ(5,208) FF
  208 FORMAT(6F6.3)
      READ(5,209) FFIL
  209 FORMAT(20F6.4)
      DO 211 I=1,3
      Q1=SQRT((2*FO(I)*A(I))/(D(I)*C))
      PIL(I)=Q1+.5
  211 PIP(I)=PIL(I)
      DO 212 I=1,3
      FOF(I)=SM(I)
      AIQ(I)=0
      BACK(I)=0
      ERT(I)=0
      IND(I)=0
      NO(I)=0
      ORD(I)=0
```

(Continued)

```
      SDF(I)=0
      KK(I)=0
      QR(I)=0
      TR(I)=3000
  212 RCQ(I)=0
      QH(1)=0
      QH(2)=0
      ACC(1)=0
      ACC(2)=0
      EM(1)=TSTA+1
      EM(2)=TERM
      DO 214 I=3,5
      K=I-2
  214 EM(I)=1+DELT(K)
      DO 215 I=6,8
  215 EM(I)=3000
      EM(9)=1
      EM(10)=1
      EM(11)=3000
      EM(12)=3000
      UD=0
      KZY=0
      KONT1=1
      KONT2=1
      KONT3=1
      MARK1=182
      MARK2=180
      MARK3=180
      JES1=0
      JES2=0
      JES3=0
      KLOK1=1
      KLOK2=1
      KLOK3=1
      NEVENT=1
      TIME=1
  216 TLAS=TIME
      CALL NEXT(NEVENT,TIME)
      IF(TIME.EQ.TLAS) GO TO 218
      KZY=KZY+1
      DO 217 I=1,3
      ARRAY1(KZY,I)=PIL(I)
  217 ARRAY2(KZY,I)=PIP(I)
  218 GO TO (220,280,230,230,240,250,250,250,260,260,270,270),NEVENT
  220 WRITE(6,221) AIQ,BACK,NO
  221 FORMAT(/,3(I8,4X))
      EM(1)=EM(1)+3000
      GO TO 216
```

(Continued)

```
230 MUTT=KONT1*MARK1+1
    IF(TIME.LE.MUTT) GO TO 231
    KONT1=KONT1+1
    JES1=JES1+MARK1
231 KLOK1=TIME-JES1
    CALL FORE1(ERT,SMAD,SDF,IND,RP,EM,SM,F,R,FO,FOF)
    I=NEVENT-2
    KK(I)=KK(I)+1
    J=KK(I)
    ARRAY3(J,I)=RP(I)
    ARRAY4(J,I)=FOF(I)
    ARRAY5(J,I)=SMAD(I)
    GO TO 216
240 MUTT=KONT2*MARK2+1
    IF(TIME.LE.MUTT) GO TO 241
    KONT2=KONT2+1
    JES2=JES2+MARK2
241 KLOK2=TIME-JES2
    CALL FORE2(ERT,SMAD,SDF,IND,RP,EM,SM,FF,R,FO,FOF)
    KK(3)=KK(3)+1
    J=KK(3)
    ARRAY3(J,3)=RP(3)
    ARRAY4(J,3)=FOF(3)
    ARRAY5(J,3)=SMAD(3)
    GO TO 216
250 CALL RCPT(PIL,RCQ,QR,EM,TR,QH)
    GO TO 216
260 IOU=NEVENT-8
    JZ=NO(IOU)
    MUTT=KONT3*MARK3+1
    IF(TIME.LE.MUTT) GO TO 262
    KONT3=KONT3+1
    JES3=JES3+MARK3
262 KLOK3=TIME-JES3
    CALL DRO1(SDF,PIL,PIP,ORD,EM,NO)
    IF(NO(IOU).EQ.JZ) GO TO 261
    I=NO(IOU)
    ARRAY6(I,IOU)=ORD(IOU)
    ARRAY7(I,IOU)=TIME
261 GO TO 216
270 JZ=NO(3)
    CALL DRO2(PIL,PIP,EM,RCQ,QR,TR,QH,ORD,NO,SDF,UD)
    IF(NO(3).EQ.JZ) GO TO 271
    I=NO(3)
    ARRAY6(I,3)=ORD(3)
    ARRAY7(I,3)=TIME
271 GO TO 216
280 SUM1=0
```

(Continued)

```
      SUM2=0
      SUM3=0
      DO 281 MN=3,38
      SUM1=SUM1+ARRAY3(MN,3)
      SUM2=SUM2+ARRAY4(MN,3)
  281 SUM3=SUM3+ARRAY5(MN,3)
      SUM1=SUM1/36
      SUM2=SUM2/36
      SUM3=SUM3/36
      LL=0
      DO 282 MN=1,65
      IF(ARRAY6(MN,3).EQ.0) GO TO 283
      LL=LL+1
  282 CONTINUE
  283 LIFT=1
      DO 284 N=1,LL
      IF(ARRAY7(N,3).GT.90) GO TO 285
      LIFT=LIFT+1
  284 CONTINUE
  285 SUM4=0
      SUM5=0
      DO 286 N=LIFT,LL
      SUM4=SUM4+ARRAY6(N,3)
  286 SUM5=SUM5+ARRAY6(N,3)**2
      EEK=SUM4**2/(LL-LIFT+1)
      SUM5=SQRT((SUM5-EEK)/(LL-LIFT)
      SUM4=SUM4/(LL-LIFT+1)
      WRITE(6,290) TLAS,TIME,IX
  290 FORMAT(/,3X,I5,3X,I5,3X,I15)
      WRITE(6,291) ((ARRAY1(J,K),ARRAY2(J,K),K=1,3),J=1,1170)
  291 FORMAT(/,6(3X,17))
      WRITE(6,292) ((ARRAY3(J,K),ARRAY4(J,K),ARRAY5(J,K),K=1,3),J=1,167)
  292 FORMAT(/,3(1X,I5,1X,F7.2,1X,F6.2))
      WRITE(6,293) ((ARRAY6(J,K),ARRAY7(J,K),K=1,3),J=1,65)
  293 FORMAT(/,6(3X,I5))
      WRITE(6,294) SUM1,SUM2,SUM3,SUM4,SUM5
  294 FORMAT(/,2X,F8.2,2X,F8.2,2X,F7.2,2X,F8.2,2X,F7.2)
      END
```

Figure 18. Main Program, Seasonal Simulator

Use of Procedure 2 in the Seasonal Simulator. This change is accomplished in the same manner as was used in the base and trend simulators. (See page 67.)

Use of Procedure 3 in the Seasonal Simulator. In applying Procedure 3 to the seasonal simulator, it is assumed that each branch facility, while making its usual forecasts, will make an additional one over the lead time of the central warehouse. The central warehouse can then compute its forecast by adding these branch forecasts. An additional vector, FO3, is included for use in storing these values, computed by the branches, until needed by the central warehouse. The variables KLOK2 and FF are no longer needed.

Figure 19 shows the revision to Subroutine FORE1. The new statements in this subroutine give the branches the capability to compute a forecast over the lead time of the central warehouse. The proper seasonal factors are selected and stored in the vector FEO. These factors are then used to compute the value FO3(K), where K denotes the branch making the forecast. Since the lead time of the central warehouse is not evenly divisible by the forecast interval of the branches, the variable REM is provided to contain the fractional remainder, expressed in days.

```
      SUBROUTINE FORE1(ERT,SMAD,SDF,IND,RP,EM,SM,F,R,FO,FOF
C     PURPOSE - TO UPDATE FORECASTS FOR FIRST-ECHELON FACILITIES.
      DIMENSION SDF(3),FO(3),ERT(3),SMAD(3),ALO(3),AHI(3),CNTR(3),IND(3)
     A,RP(3),TAU(3),DELT(3),EM(12),SM(3),R(3),F(13,2),WHI(3),WLO(3),FOF(
     B3),FO3(2),FEO(5)
      COMMON DUMM(27),TIME,NEVENT,TLAS,ALO,AHI,CNTR,TAU,DELT,DU(13),XXX
      COMMON/COM3/WHI,WLO,KLOK1,FO3
      INTEGER EM,TAU,TIME,DELT
      INTEGER SDF,RP
      K=NEVENT-2
      SUM=SDF(K)
      ERR=SUM-FOF(K)
      ERT(K)=ERT(K)+ERR
```

(Continued)

```
      SMAD(K)=ALO(K)*ABS(ERR)+(1-ALO(K))*SMAD(K)
      TS=ABS(ERT(K))/SMAD(K)
      LAMP=KLOK1+DELT(K)
      NAP=KLOK1+TAU(K)
      KK=1
      DO 30 I=15,183,14
      IF(KLOK1.LE.I) GO TO 31
   30 KK=KK+1
   31 L=KK
      N=KK
      IGG=14*KK+1
      IF(LAMP.GT.IGG) L=L+1
      IF(NAP.GT.IGG) N=N+1
      IF(L.GT.13) L=L-13
      IF(N.GT.13) N=N-13
C     THE FOLLOWING STATEMENTS SELECT THE PROPER SEASONAL FACTORS
C     TO USE IN COMPUTING FORECASTS OVER CENTRAL WAREHOUSE LEAD TIME.
      KEX=TAU(3)/DELT(K)
      J=KEX+1
      MAC=KEX*DELT(K)
      REM=TAU(3)-MAC
      MIKE=LAMP
      DO 37 I=1,J
      LL=KK
      DO 35 II=1,J
      IGG=14*LL+1
      IF(MIKE.LE.IGG) GO TO 36
   35 LL=LL+1
   36 IF(LL.GT.13) LL=LL-13
      FEO(I)=F(LL,K)
   37 MIKE=MIKE+DELT(K)
      IF(TS.LE.CNTR(K)) GO TO 32
      IND(K)=IND(K)+1
      IF(IND(K).LE.1) GO TO 33
      ERT(K)=0
      SP=SM(K)
      SM(K)=AHI(K)*SDF(K)/F(KK,K)+(1-AHI(K))*(SP+R(K))
      R(K)=WHI(K)*(SM(K)-SP)+(1-WHI(K))*R(K)
      GO TO 34
   32 IND(K)=0
   33 SP=SM(K)
      SM(K)=ALO(K)*SDF(K)/F(KK,K)+(1-ALO(K))*(SP+R(K))
      R(K)=WLO(K)*(SM(K)-SP)+(1-WLO(K))*R(K)
   34 FOF(K)=(SM(K)+R(K))*F(L.K)
      FO(K)=FOF(K)+(SM(K)+2*R(K))*F(N,K)
      XYZ=FO(K)+XXX*1.25*SMAD(K)*TAU(K)/DELT(K)
      RP(K)=XYZ+.5
C     FO3 IS FORECAST OF DEMAND AT BRANCH K OVER CENTRAL WAREHOUSE
C     LEAD TIME.
```

(Continued)

```
      FO3(K)=0
      DO 38 I=1,KEX
   38 FO3(K)=FO3(K)+(SM(K)+I*R(K))*FEO(I)
      FO3(K)=FO3(K)+((SM(K)+J*R(K))*FEO(J))*REM/DELT(K)
      SDF(K)=0
      EM(NEVENT)=TIME+DELT(K)
      RETURN
      END
```

Figure 19. Subroutine FORE1 (Revised), Seasonal Simulator

Figure 20 shows the latest revision to Subroutine FORE2.

```
      SUBROUTINE FORE2(SMAD,RP,EM,FO,FO3,FOF)
C     PURPOSE - TO UPDATE FORECASTS AND SET THE REORDER POINT
C     FOR THE CENTRAL WAREHOUSE.
      DIMENSION FO(3),SMAD(3),RP(3),TAU(3),DELT(3),EM(12),FOF(3),FO3(2),
     1VAR(2)
      COMMON DUMMY(27),TIME,DUM(11),TAU,DELT,DUMST(13),XXX
      INTEGER RP,EM,TAU,DELT,TIME
      FO(3)=FO3(1)+FO3(2)
      FOF(3)=FO(3)
      V3=0
      DO 40 I=1,2
      VAR(I)=(1.25*SMAD(I))**2
   40 V3=V3+VAR(I)*DELT(3)/DELT(I)
      SMAD(3)=.8*SQRT(V3)
      XYZ=FO(3)+XXX*1.25*SMAD(3)*TAU(3)/DELT(3)
      RP(3)=XYZ+.5
      EM(5)=TIME+DELT(3)
      RETURN
      END
```

Figure 20. Subroutine FORE2 (Revised), Seasonal Simulator

In this subroutine, the central warehouse forecast is simply

FO(3)=FO3(1)+FO3(2)

Changes in the main program are minimal. The vector FO3 is added to the DIMENSION statement and replaces KLOK2 in the COMMON statement, COM3. The section with label 240 is simplified to the following:

```
240 CALL FORE2(SMAD,RP,EM,FO,FO3,FOF)
    KK(3)=KK(3)+1
    J=KK(3)
    ARRAY3(J,3)=RP(3)
    ARRAY4(J,3)=FOF(3)
    ARRAY5(J,3)=SMAD(3)
    GO TO 216
```

Model Validation

Validation of the simulation models was accomplished in two steps. First, the subroutines were tested individually with hand-calculated data. The data used in these validation tests were sufficiently varied that the full range of subroutine capabilities was examined. Once these individual subroutines were known to be functioning properly, they were combined with the main programs, and full system output was spot-checked for consistency.

A Chi-square Goodness of Fit Test was used to evaluate the performance of the normal random variable generator, since the variables being generated are rounded to integers. A sample size of 1000 was used, and the hypothesis that the integer variables generated had the desired theoretical mean and variance was accepted at a 5 per cent level of significance.

Limitations of the Models

The simulators which use a constant or trend forecast model are quite general in design. Parameter values may be adjusted freely without having to make changes in the programs. The programs which use the seasonal model are not nearly so flexible. The six-month cycle is built into the simulators. The tables of seasonal factors are based on this cycle length and the selected forecast intervals. Thus, changes in cycle length or forecast interval will require an adjustment in the size of these tables. The forecasting subroutines are, in turn, designed around the tables of seasonal factors.

Expansion of these simulators to include additional branch facilities would require only minor changes. Addition of a third echelon, however, would require almost a complete re-write.

Lead times are assumed to be fixed in these models, but with the addition of another process generator, this assumption could be relaxed. Additional vectors would have to be provided to store shipments in transit, since the possibility of a crossing of shipments would then exist.

APPENDIX B

FORECASTING STATISTICS

In this appendix, forecasting statistics from the 42 simulation tests are presented in Tables 17-31. These statistics are average values computed for the full three years of simulation. Each table contains the values produced by all three second-echelon forecasting procedures, for a particular combination of forecasting model and demand process.

Table 17. Forecasting Statistics, Low Variance Demands, Constant Model

Procedure	Reorder Points	Forecasts	MAD
1	1720.44	1190.68	352.85
2	1217.86	1196.39	14.37
3	1218.83	1195.75	15.28

Table 18. Forecasting Statistics, High Variance Demands, Constant Model

Procedure	Reorder Points	Forecasts	MAD
1	1754.22	1203.49	367.13
2	1303.31	1207.82	63.68
3	1306.86	1207.30	66.35

Table 19. Forecasting Statistics, Dissimilar Demands, Constant Model

Procedure	Reorder Points	Forecasts	MAD
1	1290.89	892.85	265.37
2	984.39	898.61	57.14
3	970.64	895.02	50.42

Table 20. Forecasting Statistics, Similar Demands, High Level, Constant Model

Procedure	Reorder Points	Forecasts	MAD
1	1980.81	1437.08	362.44
2	1537.47	1441.70	63.88
3	1546.89	1439.66	71.47

Table 21. Forecasting Statistics, Similar Demands, Low Level, Constant Model

Procedure	Reorder Points	Forecasts	MAD
2	383.17	362.08	13.98
3	379.64	361.49	12.15

Table 22. Forecasting Statistics, Low Variance Demands, Trend Model

Procedure	Reorder Points	Forecasts	MAD
1	2792.94	2289.62	335.51
2	2281.69	2255.14	17.68
3	2273.75	2248.23	17.02

Table 23. Forecasting Statistics, High Variance Demands, Trend Model

Procedure	Reorder Points	Forecasts	MAD
1	2961.44	2306.22	436.78
2	2366.53	2273.34	62.09
3	2367.92	2267.35	67.03

Table 24. Forecasting Statistics, Dissimilar Demands, Trend Model

Procedure	Reorder Points	Forecasts	MAD
1	2175.19	1717.35	305.28
2	1783.33	1691.27	61.36
3	1776.00	1687.83	58.77

Table 25. Forecasting Statistics, Similar Demands, High Level, Trend Model

Procedure	Reorder Points	Forecasts	MAD
1	3299.08	2719.32	386.49
2	2818.22	2685.20	88.69
3	2790.14	2670.27	79.96

Table 26. Forecasting Statistics, Similar Demands, Low Level, Trend Model

Procedure	Reorder Points	Forecasts	MAD
2	695.61	675.09	13.69
3	691.89	672.07	13.21

Table 27. Forecasting Statistics, Low Variance Demands, Seasonal Model

Procedure	Reorder Points	Forecasts	MAD
1	2934.58	2342.97	394.42
2	2301.92	2252.67	32.85
3	2330.17	2258.92	47.45

Table 28. Forecasting Statistics, High Variance Demands, Seasonal Model

Procedure	Reorder Points	Forecasts	MAD
1	2967.08	2315.45	434.35
2	2384.00	2283.28	67.20
3	2419.08	2290.08	86.02

Table 29. Forecasting Statistics, Dissimilar Demands, Seasonal Model

Procedure	Reorder Points	Forecasts	MAD
1	2162.39	1693.47	312.60
2	1765.56	1670.42	63.34
3	1813.65	1703.61	73.42

Table 30. Forecasting Statistics, Similar Demands, High Level, Seasonal Model

Procedure	Reorder Points	Forecasts	MAD
1	3480.03	2753.33	484.53
2	2861.33	2719.11	94.76
3	2898.81	2742.13	104.44

Table 31. Forecasting Statistics, Similar Demands, Low Level, Seasonal Model

Procedure	Reorder Points	Forecasts	MAD
2	694.17	673.70	13.60
3	706.31	677.05	19.48

BIBLIOGRAPHY

1. Box, G. E. P. and G. M. Jenkins, "Some Statistical Aspects of Adaptive Optimization and Control," *Royal Statistical Society Journal*, April 5, 1962.

2. Brown, R. G., *Smoothing, Forecasting, and Prediction of Discrete Time Series*, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1963.

3. Brown, R. G., *Decision Rules for Inventory Management*, Holt Rinehart, and Winston, New York, N. Y., 1967.

4. Clark, A. J., "A Dynamic, Single-Item, Multi-Echelon Inventory Model," *RAND*, RM-2297, 1958.

5. Clark, A. J., and H. Scarf, "Optimal Policies for a Multi-Echelon Inventory Problem," *Management Science*, 16 (1970), 4 (July), 475-490.

6. Eilon, S. and J. Elmaleh, "An Evaluation of Alternative Inventory Control Policies," *Israel Journal of Technology*, 6 (1966), Number 3, 307-315.

7. Forrester, Jay, *Industrial Dynamics*, MIT Press, Cambridge, Mass., 1961.

8. Hadley, G. and T. M. Whitin, *Analysis of Inventory Systems*, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1963.

9. Harrison, P. J., "Short Term Sales Forecasting," *Applied Statistics (UK)*, 14 (1965), Numbers 2 and 3, 102-139.

10. Harrison, P. J., "Exponential Smoothing and Short Term Sales Forecasting," *Management Science*, 13 (1967), 11 (July), 821-842.

11. Hochstaedter, D., "An Approximation of the Cost Function for a Multi-Echelon Inventory Model," *Management Science*, 16 (1970), 11 (July), 716-727.

12. Holt, C. C., F. Modigliani, J. F. Muth, and H. A. Simon, *Planning Production Inventories and Work Force*, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1960.

13. Inglehart, D. L., "An Allocation Model," *SIAM Journal of Applied Math*, 16 (1967), 2 (March), 303-373.

14. Ledley, R. S., *FORTRAN IV Programming,* McGraw Hill Book Company, New York, N. Y., 1966.

15. Magee, J. F. and D. M. Boodman, *Production Planning and Inventory Control,* McGraw Hill Book Company, New York, N. Y., 1967.

16. Meslin, S. L., "Sales Forecasting Applied to Inventory Control," *Production and Inventory Management,* 12 (1971), 1 (First Quarter), 42-51.

17. Meyer, P. L., *Introductory Probability and Statistical Applications,* Addison-Wesley, Reading, Mass., 1970.

18. Pegels, C. C., "Exponential Forecasting: Some New Variations," *Management Science,* 15 (1969), 5 (Jan), 311-315.

19. Peterson, R., "A Note on the Determination of an Optimal Forecasting Strategy," *Management Science,* 16 (1969), 4 (Dec), B165-B169.

20. Plossl, G. W. and O. W. Wight, *Production and Inventory Control,* Prentice-Hall, Inc., Englewood Cliffs, N. J., 1967.

21. Schmidt, J. W. and R. E. Taylor, *Simulation and Analysis of Industrial Systems,* Richard D. Irwin, Inc., Homewood, Ill., 1970.

22. Shakum, M. F., "An Application of Operations Research to Multi-Warehouse Inventory Control," *Journal of Industrial Engineering,* 13 (1962), 6 (Nov-Dec), 470-474.

23. Skeith, R. W., "A Multi-Location Inventory Model," *Journal of Industrial Engineering,* 19 (1968), 12 (Dec), 630-633.

24. Strassman, P. A., "Forecasting Considerations in the Design of Management Information Systems," *Management Accounting,* 46 (1965), Feb., 27-40.

25. Winters, P. R., "Exponentially Weighted Moving Averages," *Management Science,* 6 (1960), 3 (Apr), 324-342.